

319902760 UG

日 本 国 特 許 庁

PATENT OFFICE  
JAPANESE GOVERNMENT

10025 U.S. PTO  
09/730577  
12/07/00

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日

Date of Application:

1999年12月10日

出 願 番 号

Application Number:

平成11年特許願第351116号

出 願 人

Applicant(s):

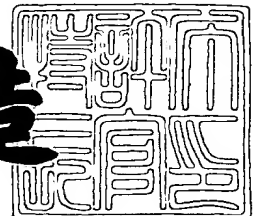
株式会社日立製作所

U.S. Appln Filed 12-7-00  
Inventor: Y. Kanada  
Mattingly Stangers malur  
Docket H-956

2000年10月 6日

特 許 庁 長 官  
Commissioner,  
Patent Office

及 川 耕 造



出証番号 出証特2000-3081639

【書類名】 特許願

【整理番号】 H99027601A

【提出日】 平成11年12月10日

【あて先】 特許庁長官 殿

【国際特許分類】 H04L 12/28  
H04L 12/56

【発明者】

【住所又は居所】 東京都国分寺市東恋ヶ窪一丁目 2 8 0 番地  
株式会社日立製作所中央研究所内

【氏名】 金田 泰

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

【代理人】

【識別番号】 100075096

【弁理士】

【氏名又は名称】 作田 康夫

【電話番号】 03-3212-1111

【手数料の表示】

【予納台帳番号】 013088

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 ネットワークポリシー転送方法および分散ルールベースプログラム転送方法

【特許請求の範囲】

【請求項 1】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のポリシーサーバが前記の複数のポリシールールに対して整数値の識別番号を付与し、  
前記のポリシーサーバからポリシールールを識別番号つきでネットワーク・ノードまたはプロキシに転送し、  
前記のポリシールールの削除要求または変更要求を前記のポリシーサーバが受信すると、  
前記のポリシーサーバから前記のネットワーク・ノードまたはプロキシに対して前記の識別番号を指定して前記の削除要求または変更要求を送信することを特徴とするネットワークポリシー転送方法。

【請求項 2】

プログラムサーバからダウンロードされる複数のルールによって構成されるプログラムによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のプログラムサーバが前記の複数のルールに対して整数値の識別番号を付与し、  
前記のプログラムサーバからルールを識別番号つきでネットワーク・ノードに転送し、  
前記のルールの削除要求または変更要求を前記のプログラムサーバが受信すると、  
前記のプログラムサーバから前記のネットワーク・ノードに対して前記の識別番号を指定して前記の削除要求または変更要求を送信することを特徴とする分散ル

ールベースプログラム転送方法。

【請求項 3】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシーをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のポリシーサーバから第 1 のポリシールールを識別子つきでネットワーク・ノードまたはプロキシーに転送し、  
前記のポリシーサーバが前記の第 1 のポリシールールと依存関係がある第 2 のポリシールールの追加要求、削除要求または変更要求を受信した際に、  
前記の第 1 のポリシールールの識別子を前記の第 2 のポリシールールの識別子とあわせて前記のネットワーク・ノードまたはプロキシーに転送することを特徴とするネットワークポリシー転送方法。

【請求項 4】

プログラムサーバからダウンロードされる複数のルールによって構成されるプログラムによって制御される複数のプロキシーをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のプログラムサーバから第 1 のルールを識別子つきでネットワーク・ノードまたはプロキシーに転送し、  
前記のプログラムサーバが前記の第 1 のルールと依存関係がある第 2 のルールの追加要求、削除要求または変更要求を受信した際に、  
前記の第 1 のルールの識別子を前記の第 2 のルールの識別子とあわせて前記のネットワーク・ノードまたはプロキシーに転送することを特徴とする分散ルールベースプログラム転送方法。

【請求項 5】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシーをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のポリシーへのポリシールールの追加要求、前記のポリシーからのポリシールールの削除要求、または前記のポリシーにふくまれるポリシールールの変更要

求を前記のポリシーサーバが受信した際に、  
前記の複数のポリシールール間に依存関係がある複数のポリシールールの識別子のすべてをネットワーク・ノードまたはプロキシに転送することを特徴とするネットワークポリシー転送方法。

【請求項 6】

プログラムサーバからダウンロードされる複数のルールによって構成されるプログラムによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のプログラムへのルールの追加要求、前記のプログラムからのルールの削除要求、または前記のプログラムにふくまれるルールの変更要求を前記のプログラムサーバが受信した際に、  
前記の複数のルール間に依存関係がある複数のルールの識別子のすべてをネットワーク・ノードまたはプロキシに転送することを特徴とする分散ルールベースプログラム転送方法。

【請求項 7】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のポリシーへの第 1 のポリシールールの追加要求または前記のポリシーにふくまれる第 1 のポリシールールの変更要求を前記のポリシーサーバが受信した際に、  
前記の第 1 のポリシールールに類似した第 2 のポリシールールがすでにネットワーク・ノードまたはプロキシに格納されていれば、  
前記の第 1 のポリシールールが前記の第 2 のポリシールールとことなる部分だけをふくむメッセージを前記のネットワーク・ノードまたはプロキシに転送することを特徴とするネットワークポリシー転送方法。

【請求項 8】

プログラムサーバからダウンロードされる複数のルールによって構成されるプログラムによって制御される複数のプロキシをともなうまたはともなわないネ

ットワーク・ノードによって構成されるネットワークにおいて、  
前記のプログラムへの第 1 のルールの追加要求または前記のプログラムにふくまれる第 1 のルールの変更要求を前記のプログラムサーバが受信した際に、  
前記の第 1 のルールに類似した第 2 のルールがすでにネットワーク・ノードまたはプロキシーに格納されていれば、  
前記の第 1 のルールが前記の第 2 のルールとことなる部分だけをふくむメッセージを前記のネットワーク・ノードまたはプロキシーに転送することを特徴とする分散ルールベースプログラム転送方法。

【請求項 9】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシーをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のポリシーサーバまたはプロキシーが受信した第 1 のポリシールールを第 2 のポリシールールと第 3 のポリシールールとをふくむ複数個のポリシールールに分割し、  
前記の第 2 のポリシールールにおいては前記の第 1 のポリシールールには出現しない特定の値をラベル変数に代入し、  
前記の第 3 のポリシールールにおいては前記のラベル変数の値を参照し、  
前記のポリシーサーバまたはプロキシーは前記の複数個のポリシールールを前記のネットワーク・ノードに送信することを特徴とするポリシールール転送方法。

【請求項 10】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシーをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記のポリシーサーバまたはプロキシーが第 1 のポリシールールと第 2 のポリシールールとをふくむ複数個のポリシールールを受信し、  
前記の第 1 のポリシールールにおいては特定の値をラベル変数に代入し、  
前記の第 2 のポリシールールにおいては前記のラベル変数の値を参照しているばあいに、

前記のポリシーサーバまたはプロキシにおいて前記の複数のポリシールールを、前記のラベル変数が出現しない 1 個の第 3 のポリシールールに変換し、前記の第 3 のポリシールールを前記のネットワーク・ノードに送信することを特徴とするポリシールール転送方法。

#### 【請求項 1 1】

プログラムサーバからダウンロードされる複数のルールによって構成されるプログラムによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、前記のプログラムサーバまたはプロキシが受信した第 1 のルールを第 2 のルールと第 3 のルールとをふくむ複数のルールに分割し、前記の第 2 のルールにおいては前記の第 1 のルールには出現しない特定の値をラベル変数に代入し、前記の第 3 のルールにおいては前記のラベル変数の値を参照し、前記のプログラムサーバまたはプロキシは前記の複数のルールを前記のネットワーク・ノードに送信することを特徴とする分散ルールベースプログラム転送方法。

#### 【請求項 1 2】

プログラムサーバからダウンロードされる複数のルールによって構成されるプログラムによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、前記のプログラムサーバまたはプロキシが第 1 のルールと第 2 のルールとをふくむ複数のルールを受信し、前記の第 1 のルールにおいては特定の値をラベル変数に代入し、前記の第 2 のルールにおいては前記のラベル変数の値を参照しているばあい、前記のプログラムサーバまたはプロキシにおいて前記の複数のプログラムルールを、前記のラベル変数が出現しない 1 個の第 3 のルールに変換し、前記の第 3 のルールを前記のネットワーク・ノードに送信することを特徴とする分散ルールベースプログラム転送方法。

【請求項 1 3】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記の複数のポリシールールの条件部における条件が排他であるかどうかを前記のポリシーサーバまたはネットワーク・ノードまたはプロキシにおいて判定し、  
排他でないときはその旨をポリシールールを入力者に報告することを特徴とするネットワークポリシー転送方法。

【請求項 1 4】

ポリシーサーバからダウンロードされる複数のポリシールールによって構成されるポリシーによって制御される複数のプロキシをともなうまたはともなわないネットワーク・ノードによって構成されるネットワークにおいて、  
前記の複数のポリシールールをオペレータが入力または編集する際に、前記の複数のポリシールールを 1 個のテンプレートに表示し、前記の複数のポリシールール間を矢印によって結合し、前記の矢印によって結合された複数のポリシールールは同一の変数の値を定義または使用することを特徴とするネットワークポリシー入力方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、ポリシーにもとづく制御が可能なルータ等のネットワーク・ノードによって構成されたインターネット等のネットワークの QoS 等の制御、およびルールベースのプログラムを実行可能なネットワーク・ノードによって構成されたネットワークにおける分散ルールベース・プログラム実行方法に関する。

【0 0 0 2】

【従来技術】

本発明に関連する第 1 の従来技術としてネットワークのポリシー制御方式がある。ネットワークのポリシー制御方式に関しては IETF (Internet Engineerin



g Task Force) 等において議論されているが、その概説としてはつぎの文献がある。

【 0 0 0 3 】

「製品化始まったポリシー・サーバー」, 日経インターネットテクノロジー, 1999 年 6 月号, pp.144- 151。

【 0 0 0 4 】

特に、ポリシーにもとづく QoS (Quality of Service) 制御法についてはつぎの文献がある。

【 0 0 0 5 】

「White Paper - Introduction to QoS Policies」, <http://www.stardust.com/>, 1998 年。

【 0 0 0 6 】

ポリシー制御のないネットワーク・システムにおいては、ネットワーク機器の QoS 管理機能 (サービス品質管理機能) やセキュリティ管理機能などを制御する際に、ネットワーク機器ごとに設定をおこなう必要がある。しかし、ポリシー制御ネットワーク・システムにおいては、ポリシーサーバとよばれるコンピュータに設定の方針すなわちポリシーを指定することによって、少量の情報を入力するだけでネットワーク全体への設定ができるようにしている。また、時刻をこまかく指定してポリシーを変更したり、アプリケーション・プログラムからの要求に応じて動的にポリシーを変更するなど、人間のオペレータでは実現困難なネットワーク制御を可能にしている。

【 0 0 0 7 】

ポリシーは通常、ポリシールールとよばれる規則の並びとして記述される。ポリシールールは条件-動作型の規則である。すなわち、ある条件がなりたつときにとるべき動作を記述する規則である。

【 0 0 0 8 】

ポリシーをルータにダウンロードする際のプロトコルには複数の候補があるが、代表的なものとして COPS (Common Open Policy Service) プロトコルがある。COPS プロトコルは IETF (Internet Engineering Task Force) において、つ

ぎの文献によって提案されている。

【0 0 0 9】

J. Boyle 他著, The COPS (Common Open Policy Service) Protocol, draft-ietf-rap-cops-08.txt (<http://www.ietf.org/internet-drafts/draft-ietf-rap-cops-08.txt>), Internet Draft, IETF, 1999 年。

【0 0 1 0】

F. Reichmeyer 他著, COPS Usage for Policy Provisioning, draft-ietf-rap-pr-01.txt (<http://www.ietf.org/internet-drafts/draft-ietf-rap-pr-01.txt>), Internet Draft, IETF, 1999 年。

【0 0 1 1】

また、ダウンロードする際のポリシーの表現に関しては、PIB (Policy Information Base) が提案されている。その一例として、つぎの文献に記述されたものがある。

【0 0 1 2】

M. Fine 他著, Quality of Service Policy Information Base, draft-mfine-cops-pib-02.txt (<http://www.ietf.org/internet-drafts/draft-mfine-cops-pib-02.txt>), Internet Draft, IETF, 1999 年。

【0 0 1 3】

本発明に関連する第 2 の従来技術として、Differentiated Services 技術（以下 DiffServ 技術とよぶ）がある。DiffServ 技術はインターネット上での QoS (Quality of Service) すなわちサービス品質を保証するための技術である。DiffServ 技術に関してはつぎのような文献がある。

【0 0 1 4】

S. Blake 他著, An Architecture for Differentiated Services, RFC 2475, IETF, 1998 年。

【0 0 1 5】

K. Nichols 他著, A Two-bit Differentiated Services Architecture for the Internet, RFC 2638, IETF, 1999 年。

## 【0016】

DiffServ 技術においては、第 1 のネットワーク・アプリケーションからネットワークを介して第 2 のネットワーク・アプリケーションとの間で一連のパケットが通信されるとき、これらが 1 個の「フロー」すなわち一連のパケットの流れに属していると考ええる。ある IP パケットがあるフローに属しているかどうかは、IP パケット上の始点と終点の IP アドレス、プロトコル、さらにそのプロトコルが TCP または UDP であるときはそのポートを識別することによって判定することができる。

## 【0017】

第 1 のネットワーク・アプリケーションから第 2 のネットワーク・アプリケーションへのパス上には、まずネットワークへの入口のエッジ・ルータがあり、0 個またはそれ以上のコア・ルータがあって、ネットワークからの出口のエッジ・ルータがある。

## 【0018】

このとき、DiffServ 技術においては、入口のエッジ・ルータにおいて複数のフローをまとめてパケット上の DS フィールド (Differentiated Services フィールド) に特定の値によってマーキングし、それ以降はその値をもつパケットをまとめてひとつのフロー (集成フロー aggregated flow と呼ぶ。)としてあつかう。DS フィールドにふくまれる値は DSCP (Differentiated Services Code Point) とよばれる。集成フローをつくることによって、コア・ルータにおいては、DSCP だけを判定することによって集成フローごとに帯域幅やパケット転送の優先度などの QoS 条件を制御することができる。DiffServ 技術を使用することによって、フローを集成し DSCP だけで判定できるようになり、QoS 条件の制御のためのコア・ルータの負荷を軽減することができる。

## 【0019】

DiffServ におけるいくつかの DSCP の値に対しては、QoS に関する標準的なふるまい (Per-hop behavior, PHB) がきめられている。Expedited Forwarding (EF) は仮想専用線にちかいふるまいであり、IETF の RFC 2598 において規定されている。EF に対して推奨されている DSCP の値は 46 である。Assured Forwa

rding (AF) は複数のふるまいがことなるサービスを定義できる枠組みであり、IETF の RFC 2597 において規定されている。また、Best Effort (BE) は従来と互換のふるまいであり、DSCP としては 0 がわりあてられている。

#### 【0 0 2 0】

##### 【発明が解決しようとする課題】

前記ポリシールールは原則的には一括してダウンロードすることも可能であり、またルール単位でダウンロードすることも可能である。PIB においては、ポリシールールやその各部分をふくめて PIB のすべての部分に識別子がつけられている。したがって、原則的には個別のルールを指定して追加・削除・更新をおこなうことができる。しかし、現実的にはルール単位ないしそれより細かい単位で追加・削除または更新すると意図した動作がえられない場合がある。ルール単位での追加・削除・更新ができない理由は 2 つある。第 1 に、一般にはルール間に依存があり、ルール単位で追加・削除・更新すると、そのルールに依存する他のルールの意味を変化させてしまうという理由がある。第 2 に、ルール間に本来は依存がなくても、ルータにおいて複数のルールをまとめて処理しているために依存が生じる場合がある。たとえば、ポリシーサーバからダウンロードされたルールをルータにおいて実行可能な形式に変換する際には、複数のルールを融合して 1 個にしたり、逆に 1 個のルールを複数のルールに分解したりする必要がある。複数のルールを融合する場合は、そのうちの 1 個のルールが削除または更新されると、他のルールもあわせて変換しなおす必要がある。

#### 【0 0 2 1】

ルール単位での追加・削除・更新ができないという問題を回避するために、従来のポリシーサーバの中には、ポリシーを一括してダウンロードするものがある。ところが、このようなポリシー一括ダウンロード型のネットワーク・システムにおいては、ポリシールールが多数になるとダウンロードに時間がかかり、また、ルータの種類によってはその間、長時間にわたってポリシー制御が効かなくなるということも考えられる。

#### 【0 0 2 2】

また、DiffServ 技術を実現する際にも同様の問題点が発生する。前記のマー

キングや QoS 条件の制御のためにはルールベースのプログラムすなわちポリシールールが使用され、ポリシーサーバによって制御されるからである。

【 0 0 2 3 】

一方、下記の文献においては、DiffServ 技術の枠組みのなかで、うまくルールの追加、削除、更新をおこなうためには各ルールがモジュラーでなければならない、つまり自由にくみあわせられるものでなければならない、また各ルールが独立であるためにはそれがふくむ条件が排他でなければならないと主張している。

【 0 0 2 4 】

Y. Kanada 他著, SNMP-based QoS Programming Interface MIB for Routers, draft-kanada-diffserv-qospifmib-00.txt (<http://www.ietf.org/internet-drafts/draft-kanada-diffserv-qospifmib-00.txt>), IETF, 1999 年。

【 0 0 2 5 】

しかしながら、この文献では、第 1 に排他でない条件が指定されたときにどのように処理するかは示されていないし、第 2 に条件が排他でないこと以外の原因による相互依存があるときにどのような方法をとればよいかについては言及されていない。

【 0 0 2 6 】

そこで、本発明の目的は、ポリシー制御ネットワーク・システムにおいて、ポリシーサーバからポリシールールをルータに追加、削除、更新する際にポリシーサーバから転送されるルールやデータを減らすことにある。

【 0 0 2 7 】

【課題を解決するための手段】

前記の課題は、つぎのような手段によって解決することができる。ポリシールール間のデータの依存性を解析する手段を使用することにより、ルータにおいてポリシールールを実行可能な形式に変換する際に変換すべき最小限のポリシールールやデータのセットを求める。また、ポリシーサーバからルータに対してポリシールールを指定する際には、当該ポリシールールがルータに格納されているかどうかを判定することにより、当該ポリシールールの内容をルータに転送せず

にその識別子だけを転送する。これにより、転送すべきデータ量を最小限にすることができる。したがって、本発明によれば、ネットワークの混雑を抑えること、ダウンロード時間およびポリシールールの変換にかかる時間を最小限にすること、ポリシー制御が中断することがないかまたは中断時間が最小限で済むようにすること、及びルータに過大な負荷がかかることがないようにすることが可能となる。

#### 【0028】

##### 【発明の実施の形態】

本発明の実施例を説明する。

#### 【0029】

まず、この実施例におけるネットワーク構成を図 1 を使用して説明する。このネットワークにおいてはインターネット・プロトコルを使用する。ネットワークはルータ 101, ルータ 111, ルータ 121 をファスト・イーサネット等によって接続することによって構成される。ルータ 101, ルータ 111, ルータ 121 をポリシーサーバ 103 が制御する。このネットワークにはアプリケーションサーバ 131, アプリケーションサーバ 132, およびこれらを利用するクライアント 141, クライアント 142 が接続されている。これにより、たとえば MPEG 画像や音声の再生, マルチメディア・データをふくむ World Wide Web の利用などをおこなう。

#### 【0030】

ルータ 101 には 192.168.1.2 という IP アドレスがあたえられている。ルータ 101 はネットワーク・インタフェース 102, 103, 104 をもっている。ネットワーク・インタフェース 102 にはインタフェース番号 1, ネットワーク・インタフェース 103 にはインタフェース番号 2, ネットワーク・インタフェース 104 にはインタフェース番号 3 が割り当てられている。

#### 【0031】

ルータ 111 には 192.168.2.2 という IP アドレスが与えられている。ルータ 111 はネットワーク・インタフェース 112, 113, 114 をもっている。ネットワーク・インタフェース 112 にはインタフェース番号 1, ネットワーク・インタ

フェース 113 にはインタフェース番号 2, ネットワーク・インタフェース 114 にはインタフェース番号 3 が割り当てられている。

【 0 0 3 2 】

ルータ 121 には 192.168.3.2 という IP アドレスが与えられている。ルータ 121 はネットワーク・インタフェース 122, 123, 124, 125 をもっている。ネットワーク・インタフェース 122 にはインタフェース番号 1, ネットワーク・インタフェース 123 にはインタフェース番号 2, ネットワーク・インタフェース 123 にはインタフェース番号 4, ネットワーク・インタフェース 124 にはインタフェース番号 5 が割り当てられている。

【 0 0 3 3 】

ルータ 101 とルータ 111 とはネットワーク・インタフェース 102 とネットワーク・インタフェース 112 とのあいだで接続されているが, このラインにはサブネット・アドレス 192.168.1.\* がわりあてられている。ルータ 111 とルータ 121 とはネットワーク・インタフェース 113 とネットワーク・インタフェース 123 との間で接続されているが, このラインにはサブネット・アドレス 192.168.2.\* が割り当てられている。ルータ 121 とルータ 111 とはネットワーク・インタフェース 122 とネットワーク・インタフェース 103 との間で接続されているが, このラインにはサブネット・アドレス 192.168.3.\* が割り当てられている。

【 0 0 3 4 】

アプリケーションサーバ 131 はルータ 101 のネットワーク・インタフェース 104 に接続されているが, この間のサブネットにはアドレス 192.168.4.\* がわりあてられている。アプリケーションサーバ 132 はルータ 111 のネットワーク・インタフェース 114 に接続されているが, この間のサブネットにはアドレス 192.168.5.\* がわりあてられている。クライアント 141 はルータ 121 のネットワーク・インタフェース 124 に接続されているが, この間のサブネットにはアドレス 192.168.6.\* が割り当てられている。クライアント 142 はルータ 121 のネットワーク・インタフェース 125 に接続されているが, この間のサブネットにはアドレス 192.168.7.\* が割り当てられている。

## 【0035】

つぎに、ポリシーサーバ 103 の構成を図 2 を使用して説明する。ポリシーサーバはパーソナル・コンピュータまたはワークステーションのような汎用のコンピュータを使用して実装される。図 2 を構成するポリシー入力処理部 202, ポリシー整合性検査部 203, ポリシールール依存関係解析部 204, ポリシースケジューリング部 205 およびポリシー送信部 206 はいずれもソフトウェアによって実現される。また、ポリシーリポジトリ 211, 変数参照表 212, ネットワーク構成管理表 213 およびポリシースケジュール表 214 はハードディスク上または主記憶上におかれる。

## 【0036】

ポリシーサーバ 103 はオペレータ・コンソール 201 と接続され、オペレータからの入力を受けつけ、またオペレータへの出力をおこなう。オペレータ・コンソール 201 を使用してポリシールールが追加、削除または更新されるが、このようなオペレータ・コンソール 201 の入出力はポリシー入力処理部 202 によって制御される。入力されたポリシールールはポリシーリポジトリ 211 に保存される。また、ポリシー入力処理部 202 によって、ポリシールールがふくむ変数参照どうしの関係が変数参照表 212 に格納される。ポリシールールはその有効期間とともに入力される。

## 【0037】

ポリシールール依存解析部 204 はポリシー入力処理部 202 から呼び出される。ポリシールール依存解析部 204 においてはポリシーリポジトリ 211 と変数参照表 212 とを参照して、追加・削除したルール間および追加・削除したルールと既存のルールとの依存関係を解析する。そして、相互に依存があるすべてのルールのルール識別子をまとめてポリシー整合性検査部 203 とポリシースケジューリング部 205 にわたす。

## 【0038】

ポリシー整合性検査部 203 においては、ポリシールール追加・削除・更新にともなうポリシールール間の整合性を検査し、不整合があればポリシー入力処理部 202 をつうじてオペレータ・コンソール 201 に表示する。



## 【0039】

ポリシースケジュールリング部 205 においては、ポリシースケジュール表 213 を使用して有効期間の開始時にポリシールールをルータに追加し、有効期間の終了時にポリシールールをルータから削除する。ポリシースケジュールリング部 205 はポリシールール依存関係解析部 204 またはポリシー送信部 206 によって起動される。ポリシールール依存関係解析部 204 からはポリシールール追加・削除・更新にともなうスケジュール変更が入力され、ポリシー送信部 206 からは送信したルールの次のスケジュールが予約される。

## 【0040】

ポリシー送信部 206 はポリシースケジュール表 213 にしたがってルータのポリシールールの追加または削除をおこなう。この際、対象となるルータを特定するためにネットワーク構成管理表 212 を使用する。

## 【0041】

つづいて、図 3 を使用してポリシー入力処理部 202 がオペレータ・コンソール 201 からうけとる入力項目について説明する。オペレータはまず入力するポリシールールの種類を選択する。この実施例においては、ポリシールールの種類として、フロー分類 (Classification)、契約違反管理 (Policing)、QoS 動作 (QoSAction)、スケジュールリング (scheduling) の 4 種類がある。この実施例においては、オペレータはテンプレート 301, 321, 341, 361, 381 に記入されているルールをこの順ですべて入力することを仮定する。

## 【0042】

オペレータが「フロー分類」を選択すると、テンプレート 301 がオペレータ・コンソール 201 に表示される。テンプレート 301 には、オペレータによってすでにフローの始点 IP アドレス、ラベル、時刻などが記入されている。テンプレート 301 はルール型として Classification 302 をふくむ。また、条件はフローのプロトコル 303、フローの始点の IP アドレス (Source IP) 304、終点の IP アドレス (Destination IP) 305、および DSCP 306 によって構成される。始点および終点の IP アドレスとしては、単純な IP アドレスだけでなく IP アドレスの範囲およびポートの範囲も指定することができる。始点の IP アドレス

としては 192.168.4.1 303 が記入されている。

【0 0 4 3】

IP アドレスの範囲を指定する際、アドレスとマスクまたはアドレスの有効ビット数によって指定することがおおい。たとえば、IP アドレスが 192.168.1.0 でマスクが 255.255.255.0 または有効ビット数が 24 ビットであれば、IP アドレスが 192.168.1.0 から 192.168.1.255 という範囲が指定される。しかし、この方法は IP アドレスの範囲を下限と上限との組によって指定するのにくらべると指定可能な範囲がせまく、とくに複数のルールがあるときにそれらの条件を排他にするのが困難なばあいが生じる。たとえば、1 個のルールにおいて 192.168.1.0 から 192.168.1.255 という IP アドレス範囲を指定し、のこりのすべての

IP アドレスに対して唯一の動作を指定しようとするとき、範囲を指定することができれば 0.0.0.0 から 192.168.0.255 までと、192.168.2.0 から 255.255.255.255 までという 2 個の範囲によって指定することができるが、マスクまたは有効ビット長だけをつかって指定しようとする、多数の範囲の or をとるかたちで指定しなければならない。したがって、効率がわるく、またそれをユーザがおこなうとすれば、非常にわかりにくいという問題点がある。IP アドレスの範囲が指定できれば、このような問題は解決される。

【0 0 4 4】

テンプレート 301 における動作は欄 307 において指定された整数値を Label という名前の変数に設定することである。ここでは VideoSource という名前がつけられた整数値が記入されている。テンプレート 301 においては、さらにこのルールの有効期間が、その開始時刻 308 と終了時刻 309 とによって指定されるようになっている。ここでは開始時刻 308 は土曜日 (Sat) であり、終了時刻 309 は日曜日 (Sun) である。これは、このルールが毎週土曜日の 0 時から日曜日の 24 時 (すなわち月曜日の 0 時) まで有効であることをあらわしている。

【0 0 4 5】

テンプレート 301 に入力されたデータがあらわすルールはつぎのような意味をもっている。IP アドレス 192.168.4.1 から送信される TCP プロトコルをつ

かったフローに対しては、毎週土曜日の 0 時から日曜日の 24 時までのあいだは VideoSource というラベルをつける。このラベルづけはつぎに起動されるルールを限定するはたらきをもっている。ただし、このラベルは、DSCP や MPLS のラベルとはちがってパケットに実際につけられるラベルではなく、ルータ内で仮想的にパケットにわりあてられるラベルである。後述するように条件部にふくまれるラベル値が VideoSource であるルールはテンプレート 311 のルールとテンプレート 321 の 1 行めのルールの 2 個なので、つぎに起動されるルールをこれらに限定している。

#### 【0046】

一方、オペレータが「契約違反管理」を選択すると、テンプレート 321 がオペレータ・コンソール 201 に表示される。テンプレート 321 には、条件としてラベル 323 と転送レートの単位 324、平均の最大転送レートに関する不等号条件 325、バースト・レートの上限に関する不等号条件 326、動作として設定されるラベル値 327、ルールの有効期間の開始時刻 328 と終了時刻 329 とが指定される。ラベル 323 としては VideoSource、転送レートの単位 324 としては kbps、最大転送レート条件 325 としては 1000 kbps をこえること、ラベル 327 としては VideoPolice が記入されている。また、開始時刻 328 としては 9 時 (9:00)、終了時刻としては 17 時 (17:00) が記入されているが、これはこのルールが毎日 9 時から 17 時まで有効であることをあらわしている。

#### 【0047】

テンプレート 321 に入力されたデータがあらわすルールはつぎのような意味をもっている。ラベルが VideoSource であるフロー（仮想フロー）に対しては、毎日 9 時から 17 時のあいだにかぎり、また平均の最大転送レートが 1000 kbps をこえるときにかぎり VideoPolice というラベルをつける。VideoSource というラベルがつけられたパケットのうち、この条件にあわないものについては、このルールは適用されず、ラベルのかきかえはおこなわれない。したがって VideoSource というラベルがついたままになる。

#### 【0048】

また、オペレータが「QoS 動作」を選択すると、テンプレート 341 がオペレ

ータ・コンソール 201 に表示される。テンプレート 341 には、条件としてラベル 343、動作として DSCP 324 のかきかえ、パケット廃棄のアルゴリズム 345、最大廃棄率 346、閾値の単位 347、最小閾値 348、最大閾値 349、ラベル 350、ルールの有効期間の開始時刻 351 と終了時刻 352 が指定される。ラベル 343 としては VideoSource、DSCP としては EF の DSCP すなわち 46、パケット廃棄のアルゴリズム 345 として WRED (Weighted Random Early Discard)、最大廃棄率 346 として 300 permil すなわち 0.3、閾値の単位 347 として packet、最小閾値 348 として 50 (packets)、最大閾値 349 として 100 (packets)、ラベル 350 として VideoSchedl、ルールの有効期間の開始時刻 351 として土曜日、終了時刻 352 として日曜日が記入されている。

#### 【 0 0 4 9 】

テンプレート 341 に入力されたデータがあらわすルールはつぎのような意味をもっている。ラベルが VideoSource であるフローに対しては、毎週土曜日 0 時から日曜日 24 時までのあいだにかぎり、DiffServ における EF (Expedited Forwarding) の動作を適用する。パケット廃棄の動作は WRED にしたがって、そのパラメタは閾値の単位 347、最小閾値 348、最大閾値 349 によって指定される。また、ラベルは VideoSchedl にかきかえる。

#### 【 0 0 5 0 】

テンプレート 361 は、テンプレートとしてはテンプレート 341 と同一であるが、入力内容だけがことになっている。ラベル 363 としては VideoPolice、パケット廃棄のアルゴリズム 365 として全廃棄、ルールの有効期間の開始時刻 371 として 9 時、終了時刻 372 として 17 時が記入されている。パケットがすべて廃棄されるため後続のルールの指定は不要であるため、ラベルの欄 370 は無指定となっている。

#### 【 0 0 5 1 】

テンプレート 361 があらわすルールはつぎのような意味をもっている。ラベルが VideoPolice であるフローに対しては、毎日 9 時から 17 時までのあいだにかぎり、パケット全廃棄の動作を適用する。

## 【0052】

オペレータが「スケジューリング」を選択すると、テンプレート 381 がオペレータ・コンソール 201 に表示される。テンプレート 381 には、条件としてラベル 383、動作としてレート単位 384、最低レート 385、最高レート 386、親スケジューリング・ラベル 387 が指定され、ルールの有効期間の開始時刻 388 と終了時刻 389 とが指定される。ラベル 383 としては VideoSched1、レート単位 384 としては kbps、最低レート 385 としては 1000 (kbps)、最高レート 386 としては 2000 (kbps)、親スケジューリング・ラベル 387 としては PrioritySched1 が指定され、開始時刻 388 としては土曜日、終了時刻 389 としては日曜日が記入されている。

## 【0053】

テンプレート 381 に入力されたデータがあらわすルールはつぎのような意味をもっている。ラベルが VideoSched1 であるフローに対しては、毎週土曜日 0 時から日曜日 24 時までのあいだにかぎり、最低レートとして 1000 kbps を保証し、最高レートとして 2000 kbps をわりあてる。最高レートをこえるトラフィックがあれば、シェイピングされ、2000 kbps 以下におさえられる。すなわち、2000 kbps をこえる入力があったときは、こえるぶんはキューに蓄積し、2000 kbps ぶんだけを出力する。それによってキューがあふれたばあいにはパケット廃棄がおこる。スケジューリング法としては優先度スケジューリングがつかわれる。優先度スケジューリングが選択される理由は図 3 には記述されていないが、PrioritySched1 というラベルをもつスケジューリング・ルールがあらかじめあたえられているためである。

## 【0054】

つぎに、以上の入力項目全体としての意味について説明する。IP アドレスが 192.168.4.1 であるアプリケーション・サーバ 131 からのフローに対しては、毎週土曜日の 0 時から日曜日の 24 時までのあいだ、DiffServ における EF のあつかいで QoS を保証する。廃棄アルゴリズムとしては WRED を使用し、最低帯域 1000 kbps を保証するが、最高レートを 2000 kbps とし、これをこえるときはすべてのパケットを廃棄する。スケジューリング法としては優先度スケジュー

ーリング（優先度にもとづくスケジューリング）を使用する。すなわち，当該パケットをふくむフローに対しては Best Effort トラフィックよりたかい優先度をあたえる。ただし，毎日 9 時から 17 時のあいだは転送レートが 1000 kbps をこえないかぎり上記のあつかいとするが，これをこえたばあいには，こえたぶんのパケットをすべて廃棄する。

## 【0055】

つづいて，図 4 を使用してポリシーリポジトリ 111 の内容について説明する。図 4 は，図 3 のすべての入力がこの順にポリシー入力部 202 にあたえられたときのポリシーリポジトリ 111 の内容をあらわしている。ルール表 401 は入力されたすべてのルールをふくむ固定長の表である。列 411 がルール識別子をあらわし，列 412 がルールの型をあらわし，列 413 がルールの条件部，列 414 がルールの動作部をあらわしている。ルール識別子とルールの型は整数値であらわされるため固定長であるが，条件部および動作部は可変長でなければ記述できないばあいがあり，そのばあいは可変長のテーブルをポインタで指示することによって表現している。

## 【0056】

行 402 においてはルール識別子 411 が #1 であるルールが記述されているが，これはテンプレート 301 によって入力されたルールである。ルールの型 412 の値は「フロー分類」(Classification) となっている。また，条件は (A) のテーブルによってあらわされている。このテーブルは，プロトコル 420 が TCP であり，フローの始点の IP アドレスが 192.168.4.1 421 でありフローの終点の IP アドレスが 192.168.4.1 422 である以外は，すべて無効なデータ null によってうめられている。さらに，動作はラベル変数に 1 という値をあたえるという動作である。

## 【0057】

行 403 においてはルール識別子 411 が 2 であるルールが記述されているが，これはテンプレート 321 によって入力されたルールである。ルールの型 412 の値は「契約違反管理」(Policing) となっている。また，条件は (B) のテーブルによってあらわされている。このテーブルにおいては，ラベル 431 が 1，平

均レート 432 が 1000 であり、バースト・レート 433 は無効なデータ null によってうめられている。さらに、動作はラベル変数に 2 という値をあたえるという動作である。

## 【0058】

行 404 においてはルール識別子 411 が 3 であるルールが記述されているが、これはテンプレート 341 によって入力されたルールである。ルールの型 412 の値は「QoS 動作」(QoSAction) となっている。また、条件はラベル変数の値が 1 だということであり、動作は (C) のテーブルによってあらわされている。このテーブルにおいては、DSCP 441 が 46、廃棄アルゴリズム 442 が WRED、最大廃棄率 443 が 300 permil、閾値の単位 444 がパケット、最小閾値 445 が 50 (packets)、最大閾値 446 が 100 (packets)、あたえるラベル値 447 が 3 である。

## 【0059】

行 405 においてはルール識別子 411 が 4 であるルールが記述されているが、これはテンプレート 361 によって入力されたルールである。ルールの型 412 の値は「QoS 動作」(QoSAction) となっている。また、条件はラベル変数の値が 2 だということであり、動作は (D) のテーブルによってあらわされている。このテーブルにおいては、DSCP 451 が無指定をあらわす 255、廃棄アルゴリズム 452 が全廃棄を意味する DropAll、それ以下の欄はすべて無効な値をあらわす null がふくまれている。

## 【0060】

行 406 においてはルール識別子 411 が 5 であるルールが記述されているが、これはテンプレート 381 によって入力されたルールである。ルールの型 412 の値は「スケジューリング動作」(Scheduling) となっている。また、条件はラベルの値が 3 だということであり、動作は (E) のテーブルによってあらわされている。このテーブルにおいては、最低保証帯域 461 が 1000 kbps、平均最大帯域 462 が 2000 kbps、上位スケジューリングを指定するラベル 464 が PrioritySched1 である。

## 【0061】

つづいて、図 5 を使用してポリシー入力処理部 202 の動作を説明する。ポリシー入力処理部 202 の動作が開始されると、ステップ 501 から 532 までの処理を無限にくりかえす。まずステップ 501 においてはルール編集メニューすなわち「新規ルール定義」か「既存ルール編集」か「ポリシー送信」(Deploy)かのいずれかを選択するメニューをオペレータ・コンソール 201 に表示して、オペレータの入力を待つ。つぎにステップ 502 においてオペレータ入力新規ルール定義、既存ルール編集、ポリシー送信のいずれであるかを判定する。新規ルール定義のばあいにはステップ 511 にすすみ、既存ルール編集のばあいにはステップ 521 にすすみ、ポリシー送信のばあいはステップ 532 にすすむ。

## 【0062】

ステップ 511 においては、現在つかわれていないルール識別子を 1 個生成する。つぎにステップ 512 において、ルールタイプ入力メニューをオペレータ・コンソール 201 に表示して、オペレータの入力を待つ。さらにステップ 514 において、前記のルール識別子をキーとしてルールのタイプ、条件部、動作部の内容をポリシーリポジトリ 211 に登録する。テンプレート 301 の内容が入力されたときには、ステップ 514 において条件部としてデータ 421 から 427 が生成され、VideoSource 307 に対応する整数値としてラベル値 1 がわりあてられ、動作部の内容とされる。そして、ポリシーリポジトリ 211 にそのルールが登録される。ステップ 531 にすすむ。

## 【0063】

ステップ 521 においては、すでに入力されたルールのなかから編集するルールを選択するためのルール選択メニューを表示して、オペレータの入力を待つ。オペレータの入力があつたら、ステップ 522 において、図 4 にしるされたいずれかのテンプレートを使用し、選択されたルールの内容と「OK」ボタンおよび「削除」ボタンを表示してオペレータによるボタンのクリックを待つ。ここでオペレータはテンプレート上のルールの内容を自由にかきかえることができる。ボタンがクリックされたら、ステップ 523 において当該ルールのルール識別子を変数参照表 212 からすべて削除する。つぎにステップ 524 において、クリックさ



れたのが「OK」ボタンなのか「削除」ボタンなのかを判定する。「OK」ボタンがクリックされたばあいはステップ 525 にすすみ、「削除」ボタンがクリックされたばあいはステップ 527 にすすむ。

#### 【0 0 6 4】

ステップ 525 においては編集されたルールのルール識別子をキーとしてポリシーポジトリ 211 に編集後のルールの内容を登録する。つぎにステップ 526 において当該ルールのルール識別子を変数参照表 212 に登録する。そして、ステップ 531 にすすむ。ステップ 527 においてはルール識別子をキーとする既登録のルールをポリシーポジトリ 211 から削除する。そして、ステップ 531 にすすむ。

#### 【0 0 6 5】

ステップ 531 においてはポリシー整合性検査部 203 をよびだす。その結果、ルール間不整合が検出されたらオペレータ・コンソール 201 に報告する。そして、ステップ 501 にもどってつぎのオペレータ入力を待つ。

#### 【0 0 6 6】

ステップ 532 においては、ポリシールール依存関係解析部 204 をよびだすことによって、ルールの追加・削除にともなって影響をうけるルールのルール識別子のリストをもとめ、そのリストと追加されたルールとをルータに送信する。

#### 【0 0 6 7】

つづいて、図 14(a) を使用して変数参照表 212 の内容を説明する。変数参照表 212 は変数定義表 1401 と変数使用表 1421 とによって構成されている。変数定義表 1401 は 3 個の要素をふくんでいる。第 1 要素 1411 はルール識別子として #1 という値をふくんでいる。これは、番号 1 の変数がルール識別子 #1 のルールにおいて定義されていることをあらわす。第 2 要素 1412 はルール識別子として #2 という値をふくんでいる。これは、番号 2 の変数がルール識別子 #2 のルールにおいて定義されていることをあらわす。第 3 要素 1413 はルール識別子として #3 という値をふくんでいる。これは、番号 3 の変数がルール識別子 #3 のルールにおいて定義されていることをあらわす。

## 【 0 0 6 8 】

変数使用表 1421 は 3 個のリストをふくんでいる。第 1 のリスト 1431 はルール識別子として #2 (1421) および #3 (1422) という値をふくんでいる。これは、番号 1 の変数がルール識別子 #2 と #3 のルールにおいて使用されていることをあらわす。第 2 のリスト 1432 はルール識別子として #4 という値をふくんでいる。これは、番号 2 の変数がルール識別子 #4 のルールにおいて使用されていることをあらわす。第 3 のリスト 1433 はルール識別子として #5 という値をふくんでいる。これは、番号 3 の変数がルール識別子 #5 のルールにおいて使用されていることをあらわす。

## 【 0 0 6 9 】

変数参照表 212 の内容は図 14(b) のグラフと等価である。図 14(b) のグラフにおいて、各ノードがルールをあらわし、ノード内の番号がルール番号をあらわしている。各有向辺の始点は値の定義をあらわし、終点は値の使用をあらわす。辺 1471 はルール識別子 #1 のルールで定義された値がルール識別子 #2 のルールにおいて使用されていることをあらわす。

## 【 0 0 7 0 】

つづいて、図 16 を使用してポリシー依存関係解析部 204 の動作を説明する。ポリシー依存関係解析部 204 の実行が開始されると、まずステップ 1601 において、変数参照表 212 を使用して、新規入力または編集された以降にまだルータに送信していないすべてのルールからの変数参照関係に関するルールの推移閉包をもとめる。ルータに送信していないすべてのルールをもとめるためには、ルール表 401 にフラグのための欄をもうけて、ルールが入力されるか編集された際にはフラグをクリアし、送信した際にはフラグをたてる処理をおこなっておけば、フラグがたっていないすべてのルールをもとめることによって実現することができる。これはルール識別子 #1, #2, #3, #4, #5 のルールのうちのいずれかまたは複数のくみあわせがあたえられたばあいについていえば、図 14(b) のグラフにおいて連結成分をもとめることにひとしい。推移閉包をもとめるアルゴリズムはつぎの文献に記述されている。

【 0 0 7 1 】

A. V. エイホ, J. E. ホップクロフト, J. D. ウルマン 著, アルゴリズムの設計と解析 I, サイエンス社, 180 ページから 182 ページ, 1977 年。

【 0 0 7 2 】

または

石畑 清 著, アルゴリズムとデータ構造, 岩波講座 ソフトウェア科学 3, 岩波書店, 275 ページから 276 ページ, 1989 年。

【 0 0 7 3 】

つぎに, ステップ 1602 において, もとめた閉包の要素をトポロジカル・ソートする。これによって, 変数の定義がつねにその使用に先行するようにルールをならべかえることができる。この実施例においては, ルール識別子が #1, #2, #4, #3, #5 または #1, #3, #5, #2, #4 のような順にならべかえられる。トポロジカル・ソートのアルゴリズムはつぎの文献に記述されている。

【 0 0 7 4 】

石畑 清 著, アルゴリズムとデータ構造, 岩波講座 ソフトウェア科学 3, 岩波書店, 242 ページから 244 ページ, 1989 年。

【 0 0 7 5 】

なお, 変数参照表 212 の内容は適当なタイミングでクリアする必要があるが, ステップ 1604 の直後はそのためのタイミングとして適切である。

【 0 0 7 6 】

つづいて, 図 6 を使用してポリシー整合性検査部 203 の動作を説明する。ポリシー整合性検査部 203 の実行が開始されると, まずステップ 600 において指定されたルールと並立するルールすべてについてステップ 601 からステップ 604 までの処理をくりかえす。指定されたルールと並立するルールをもとめるには, 指定されたルールがフロー分類ルールであればすべてのフロー分類ルールをルール表 401 をつかってもとめればよい。また, 指定されたルールが契約違反管理ルールであれば, 変数参照表 212 を使用して当該ルールが使用する変数を使用するすべてのルールのなかから契約違反管理ルールだけをすべて選択すればもとめることができる。QoS 動作ルールおよびスケジューリング・ルールに関して

は、並立するルールは存在しない。

【0077】

ステップ 601 においてはまだ処理すべきルールがあるかどうかを判定する。まだあればステップ 603 にすすみ、もうなければステップ 602 にすすむ。ステップ 602 においてはポリシーが整合していることをよびだしもとに報告して、ポリシー整合性検査部 203 の処理を終了する。ステップ 603 においては、前記のルールと新規入力ルールの条件が排他であるかどうかを判定する。排他であればステップ 601 にもどって、つぎのルールの処理をつづける。排他でなければステップ 604 にすすむ。ステップ 604 においてはポリシーが不整合であることをよびだしもとに報告して、ポリシー整合性検査部 203 の処理を終了する。

【0078】

ポリシーが不整合であるばあいは、よびだしもとであるポリシー入力処理部 202 によってオペレータ・コンソール 201 に不整合であることが表示される。この実施例においてはポリシー整合性検査部 203 において複数のルールの条件が排他であるかどうかを検査しているが、これをオペレータに報告して修正をもとめるのは、複数のルールの条件が排他でないと、第 1 にポリシーサーバ 103 やルータ 101 においてルールの順序を変更したときにルールの意味が変化してオペレータの意図にあわない動作をひきおこすことがあり、第 2 に排他でないルールどうしは一種の依存関係があるため独立にはあつかえず、したがってポリシーサーバ 103 からルータ 101 への転送量をふやし、またルータ 101 におけるポリシールールコンパイラ 1103 の負荷を増加させることになるからである。

【0079】

つづいて、図 7 を使用してポリシースケジュール表 213 の内容を説明する。図 7 は、1999 年 11 月 26 日 18 時に図 3 にしめした各テンプレートに記入されたルールをすべて入力した直後のポリシースケジュール表 213 の内容をあらわしている。表の欄 721 はルール識別子、欄 722 はスケジュールされたイベント、欄 723 はつぎにそのイベントが発生すべき次時刻、欄 724 はルールにおいて指定されたかたちの時刻指定をあらわしている。先頭項目 702 はつぎのような意味をもつ。ルール識別子は #1 である。イベントは Deploy、つまり指

定されたルールをルータに追加することをあらわす。次時刻としては 1999 年 11 月 27 日の 0 時が指定されている。時刻指定は土曜日 (Sat) である。6 番めの項目 707 はつぎのような意味をもつ。ルール識別子は #2 である。イベントは Undeploy, つまり指定されたルールをルータから削除することをあらわす。次時刻としては 1999 年 11 月 27 日の 17 時が指定されている。時刻指定は (毎日) 17 時である。

【0080】

つづいて、図 8 を使用してポリシースケジューリング部 204 の動作を説明する。ポリシースケジューリング部 204 の実行が開始されると、まずステップ 801 においてポリシールール依存解析部 204 またはポリシー送信部 205 からの入力进行待つ。ステップ 802 においてどちらからの入力であるかを判定する。ポリシールール依存解析部 204 からの入力のときはステップ 803 にすすみ、ポリシー送信部 205 からの入力のときはステップ 804 にすすむ。

【0081】

ステップ 803 においては入力したルールからイベントが Deploy であるスケジュール表項目とイベントが Undeploy であるスケジュール表項目とを生成して、ポリシースケジュール表 213 に挿入する。ルール識別子が #1 であるルールすなわちポリシーリポジトリ 211 において行 402 によって表現されているルールを処理するばあいは、イベントが Deploy であるスケジュール表項目としては項目 702, イベントが Undeploy であるスケジュール表項目としては項目 709 が生成され、ポリシースケジュール表 213 に挿入される。挿入位置は次時刻欄 723 がふくむ値が昇順になるように定められる。

【0082】

項目 702 における指定時刻欄 724 の Sat (土曜日) という値はテンプレート 301 における開始時刻 308 からコピーされ、項目 709 における指定時刻欄 724 の Sun (日曜日) という値はテンプレート 301 における終了時刻 309 からコピーされる。また、項目 702 における次時刻欄 723 の 1999 年 11 月 27 日の 0 時という値は、指定時刻欄 724 において指定された「毎週土曜日」のはじめという条件をみたす現在時刻すなわち 1999 年 11 月 26 日 18 時からもっとも

ちかい未来の時刻である。項目 709 における次時刻欄 723 の 1999 年 11 月 29 日の 0 時という値は、指定時刻欄 724 において指定された「毎週日曜日」のおわりという条件をみたす現在時刻からもっともちかい未来の時刻である。

【0083】

ステップ 804 においては、送信したスケジュール表項目からつぎの項目を生成してポリシースケジュール表 213 に挿入する。生成する項目はルール識別子欄 721、イベント欄 722、指定時刻欄 724 が送信した項目とひとしく、次時刻欄 723 が送信した項目における次時刻欄よりひとつ先の時刻をふくむ。項目 702 を送信したときは、ルール識別子欄 721 が #1、イベント欄 722 が Deploy、次時刻欄 723 が 1999-12-4 0:00 すなわち 1999 年 12 月 4 日の 0 時、指定時刻欄 724 が Sat という項目が生成され、ポリシースケジュール表 213 の末尾に追加される。

【0084】

ステップ 803 および 804 の終了後はステップ 801 にもどって、つぎの入力を待つ。

【0085】

つづいて、図 9 を使用してネットワーク構成管理表 212 の内容を説明する。ネットワーク構成管理表 212 は対象 IP アドレス欄 911、ルータ IP 欄 912、ルータ・インタフェース欄 913 という 3 個の欄によって構成されている。ネットワーク構成管理表 212 には 4 個の項目が登録されている。最初の項目 902 においては、対象 IP アドレス欄 911 が 192.168.4.\*、ルータ IP 欄 912 が 192.168.1.2、ルータ・インタフェース欄 913 が 3 である。項目 902 はサブネット 192.168.4.\* に接続されているのが IP アドレス 192.168.1.2 のルータのインタフェース番号が 3 のインタフェースであることをあらわしている。

【0086】

つづいて、図 24 を使用してポリシーサーバ 103 からルータ 101 におくられるプロトコル・データの内容を説明する。当該プロトコルにおいては、3 種類の命令データを使用する。第 1 は Deploy 命令 2401 であり、第 2 は Redeploy 命令 2431 であり、第 3 は Undeploy 命令 2441 である。

## 【 0 0 8 7 】

Deploy 命令 2401 は 1 個のルールのルール識別子、その内容とそれが作用するネットワーク・インタフェースの情報をふくむ。Deploy 命令 2401 は当該ルールを送信先のルータに格納し、指定したネットワーク・インタフェースに作用させることを指示する。ルータのOp コード 2402 は Deploy という値をふくんでいるが、これはこのデータが Deploy 命令をあらわしていることをしめす。ルール識別子欄 2403 は当該 Deploy 命令がふくむルールのルール識別子をあらわす。インタフェース欄 2404 は当該 Deploy 命令が作用するべきネットワーク・インタフェースの番号をあらわす。

## 【 0 0 8 8 】

条件部のながさ 2412 は当該 Deploy 命令がふくむルールの条件部のながさをバイトを単位としてあらわす。Deploy 命令 2401 においては 32 という値がふくまれているが、これはプロトコル欄 2413 から DSCP 欄 2420 まで、条件部が 32 バイトあることをしめす。条件部には TCP という値をふくむプロトコル欄 2413, 192.168.4.1 という値をふくむフロー始点 IP アドレス下限欄 2414, 192.168.4.1 という値をふくむフロー始点 IP アドレス上限欄, null という無効な値をふくむフロー始点ポート欄 2416, フロー終点アドレス下限欄 2417, フロー終点アドレス上限欄 2418, フロー終点ポート欄 2419, DSCP 欄 2420 がふくまれている。

## 【 0 0 8 9 】

また、動作部のながさ 2421 は当該 Deploy 命令がふくむルールの動作部のながさをバイトを単位としてあらわす。Deploy 命令 2401 においては 12 という値がふくまれているが、これはラベル値欄 2422 からバーストレート欄 2424 まで、動作部が 12 バイトあることをしめす。動作部には 1 という値をふくむラベル値欄 2422, 1000 (kbps) という値をふくむ最高レート欄 2423, null という値をふくむバーストレート欄 2424 がふくまれている。

## 【 0 0 9 0 】

Redeploy 命令 2431 は 1 個のルールのルール識別子とそれが作用するネットワーク・インタフェースの情報をふくむ。Redeploy 命令 2431 は送信先のルー

タに格納されているはずの当該ルールを、あわせて送信する他の命令とともに指定したネットワーク・インタフェースに作用させることを宣言する。Op コード 2432 は Redeploy という値をふくんでいるが、これはこのデータが Redeploy 命令をあらわしていることをしめす。ルール識別子欄 2433 は当該 Redeploy 命令がふくむルールのルール識別子をあらわす。インタフェース欄 2434 は当該 Redeploy 命令が作用すべきネットワーク・インタフェースの番号をあらわす。

## 【0091】

Undeploy 命令 2441 は 1 個のルールのルール識別子とそれが作用するネットワーク・インタフェースの情報をふくむ。Undeploy 命令 2441 は、当該ルールが指定したネットワーク・インタフェースに作用しないようにすることを指示し、あわせて、当該ルールがいずれのネットワーク・インタフェースにおいてもつかわれなくなったときは、当該ルールをルータから削除してもよいことを宣言する。実際に当該ルールをルータから削除するかどうかの判断は、送信先のルータにまかせられる。Op コード 2442 は Undeploy という値をふくんでいるが、これはこのデータが Undeploy 命令をあらわしていることをしめす。ルール識別子欄 2443 は当該 Undeploy 命令がふくむルールのルール識別子をあらわす。インタフェース欄 2444 は当該 Undeploy 命令が作用すべきネットワーク・インタフェースの番号をあらわす。

## 【0092】

つづいて、図 10 を使用してポリシー送信部 205 の動作を説明する。ポリシー送信部 205 の実行が開始されると、まずステップ 1001 においてポリシースケジュール表 213 の先頭項目をポップする。すなわち、先頭項目をとりだして、ポリシースケジュール表 213 からは削除する。つぎに、ステップ 1002 においてネットワーク構成管理表 212 から、とりだしたスケジュール表項目に対応するルータとそのインタフェース番号をもとめる。項目 702 をポップしたときは、ルール識別子が #1 なのでポリシーリポジトリ 211 をルール識別子として #1 を指定して検索し、もとめた項目 402 の条件欄 413 から指示されているテーブル (A) の始点 IP アドレス 192.168.4.1 によってネットワーク構成管理表 212 をひく。IP アドレス 192.168.4.1 をふくむサブネットは 192.168.4.\* な



ので項目 902 が結果としてえられる。したがって、ルータの IP アドレス 192.168.1.2 とインタフェース番号 3 がもとめられる。

#### 【0093】

つづいてステップ 1003 において前記のスケジュール項目において指定された時刻まで待つ。項目 702 の処理においては 1999 年 11 月 27 日の 0 時にステップ 1004 以下が実行される。ステップ 1004 においてはスケジュール項目において指定されたイベントがなんであるかを判定する。それが Deploy であればステップ 1011 にすすみ、Undeploy であればステップ 1021 にすすむ。

#### 【0094】

ステップ 1011 においてはスケジュール項目において指定されたルールがすでに前記のルータに送信済みであり、前記のルータに格納されているかどうかを判定する。ここでは、前記のルータにといあわせるのではなく、ポリシーサーバ 103 がもつ情報によって判定をおこなう。図 16 の説明でのべたようにルールを送信したかどうかをフラグで管理していれば、ステップ 1011 においてはこのフラグを参照するだけで判定をおこなうことができる。判定の結果、まだ前記のルータ上にルールが存在すると判定されればステップ 1012 にすすみ、存在しないと判定されればステップ 1016 にすすむ。

#### 【0095】

ステップ 1012 においては、前記のルータに前記のルールに関する redeploy 命令を送信する。この redeploy 命令には、前記のルールのルール識別子と前記のインタフェース番号とを指定する。そして、ステップ 1031 にすすむ。

#### 【0096】

ステップ 1016 においては、前記のルータに前記のルールに関する deploy 命令を送信する。この deploy 命令には、前記のルールのルール識別子とルールの内容、そして前記のインタフェース番号を指定する。項目 702 の処理においては、指定されたルールのルール識別子が #1 なので、ポリシーリポジトリ 211 から項目 402 および項目 402 から指示されるテーブル (A) をとりだして、その内容を送信する。命令の種類は項目 702 のイベント欄 722 において指定されている deploy である。そして、ステップ 1031 にすすむ。

## 【0097】

ステップ 1016 はルールが追加されたときと更新されたときの両方のばあいに行われる。ルータ 1011 に対して、追加されたときは前記のルールを追加し、更新されたときは前記のルールと同一のルール識別子ですでに定義されていたルールを前記のルールによって置換することになる。

## 【0098】

ステップ 1021 においては、前記のルータに前記のルールに関する `undeploy` 命令を送信する。この `undeploy` 命令には、前記のルールのルール識別子と前記のインタフェース番号とを指定する。最後にステップ 1031 においてポリシースケジューリング部 205 を起動する。

## 【0099】

ルータ 101 の構成を図 11 を使用して説明する。ルータ 111、ルータ 121 の構成も図 11 のとおりである。以下で説明するポリシー受信部 1101、ポリシールール依存解析部 1102、ポリシールール・コンパイラ 1103 はソフトウェアによって実現される。また、クロスバースイッチ 1120、ネットワーク・インタフェース 1122、ネットワーク・インタフェース 1123 はハードウェアによって実現される。トラフィック制御部 1121、ルーティング制御部 1124 はソフトウェアまたはハードウェアによって実現される。変数参照表 1112 とポリシーソースルール DB 1111 は主記憶または他の半導体記憶上におかれる。ポリシールール表 1113 とキュー設定表 1114 はレジスタまたは主記憶上におかれる。

## 【0100】

ポリシー受信部 1101 はポリシーサーバ 103 からポリシールールを受信してポリシーソースルール DB 1111 に格納し、変数参照表データを受信して変数参照表 1112 に格納する。さらに、受信したルール識別子リストをポリシールールコンパイラ 1103 にわたす。ポリシールールコンパイラ 1103 はうけとったルール識別子リストにふくまれるルールを一括して変換し、結果をポリシールール表 1113 およびキュー設定表 1114 に格納する。

## 【0101】

トラフィック制御部 1121 はポリシールール表 1113 とキュー設定表 1114 を

使用してネットワーク・インタフェース 1122 およびネットワーク・インタフェース 1123 におけるネットワーク・トラフィックを制御する。クロスバースイッチ 1120 はネットワーク・インタフェース間のデータ転送をおこない、それをルーティング制御部 1124 が制御する。

【 0 1 0 2 】

つづいて図 12 を使用してネットワーク・インタフェース 1122 の構成を説明する。ネットワーク・インタフェース 1123 の構成も図 12 のとおりである。ネットワーク・インタフェース 1122 に入力されたパケットはまずフロー分類部 1201 においてどのフローに属するか分類される。フロー分類ルールはフロー分類部 1201 を制御する。つぎにフロー計測部において前記のフローが指定されたトラフィック条件をみたしているかどうかを判定し、その結果にしたがってスケジューリング部 1203 において、スケジューリング部がふくんでいる出力キューのなかから当該パケットをいれるキューを選択し、必要に応じてフローをシェイピングし、パケットを廃棄するなどの動作をとる。キューからの出力はクロスバースイッチ 1120 におくりだされる。

【 0 1 0 3 】

つづいて、図 13 を使用してポリシーソースルール DB 1111 の内容を説明する。図 13 は図 3 のすべての入力がこの順にポリシー入力部にあたえられたときのポリシーソースルール DB 1111 の内容をあらわしている。ポリシーソースルール DB 1111 のルール識別子欄 1311, ルール型欄 1312, 条件欄 1313, 動作欄 1314 の内容はポリシーリポジトリ 211 のルール識別子欄 411, ルール型欄 412, 条件欄 413, 動作欄 414 とひとしい。インタフェース欄 1316 は当該ルールを作用させるべきネットワーク・インタフェースの番号をあらわす。また、コード欄 1317 はポリシールール表 1113 において当該ルールを変換して格納した番地をあらわす。

【 0 1 0 4 】

行 1302 にはルール識別子 1311 が 1 であるルールが格納されているが、これはテンプレート 301 によって入力されたルールである。インタフェース欄 1316 は 3, コード欄は 90 をふくんでいる。行 1303 にはルール識別子 1311 が

2 であるルールが格納されているが、これはテンプレート 321 によって入力されたルールである。インタフェース欄 1316 は 3、コード欄 1317 は 90 をふくんでいる。コード欄の値が行 1302 とひとしいのは、実行可能な形式としては行 1302 のルールと行 1303 のルールとが 1 個のルールのなかにまとめて表現されていることをあらわしている。行 1304、行 1305 のルールも、実行可能な形式としては同一のルールのなかにまとめて表現されている。

#### 【0 1 0 5】

つづいて、図 15 を使用してポリシー受信部 1101 の動作を説明する。ポリシー受信部 1101 の実行が開始されると、まずステップ 1501 においてポリシーサーバ 101 からの送信データを待つ。データを受信したら、ステップ 1502 において受信したデータがふくむ命令がなんであるかを判定する。Deploy 命令であるばあいはステップ 1511 にすすみ、Undeploy 命令であるばあいはステップ 1521 にすすみ、Redeploy 命令であるばあいはステップ 1512 にすすむ。

#### 【0 1 0 6】

ステップ 1511 においては、Deploy 命令にふくまれるルールをポリシーソースルール DB にルール識別子をキーとして登録する。そして、ステップ 1512 にすすむ。

#### 【0 1 0 7】

ステップ 1512 においては、受信したルールのルール識別子を変数参照表 1112 に登録する。そして、ステップ 1531 にすすむ。

#### 【0 1 0 8】

ステップ 1521 においては、Undeploy 命令において指定されたルール識別子をもつルールを削除する。つづいて、ステップ 1522 において、受信したルールのルール識別子をすべて変数参照表 1112 から削除する。そして、ステップ 1531 にすすむ。

#### 【0 1 0 9】

ステップ 1531 においては、継続して受信するデータがあるかどうかを判定する。データがあるときはステップ 1501 にもどってつぎの受信データを処理する。データがないときはステップ 1532 にすすむ。すなわち、ステップ 1531 にお

いては一定時間内につぎのデータが到着するかどうかをしらべ、到着すればそれを処理し、到着しなければつぎの処理にうつる。ステップ 1532 においては、継続して受信したすべてのルール識別子を指定してポリシールール依存関係解析部 1102 をよぶ。そして、ステップ 1501 にもどってつぎの受信データを処理する。

#### 【 0 1 1 0 】

ステップ 1531 においては一定時間内につぎのデータが到着するかどうかによってひとかたまりのデータを識別しているが、より確実に高速にひとかたまりのデータを識別するためには、ポリシーサーバ 103 からの送信データのなかにデータのくぎりをあらわす命令をいれればよい。すなわち、Commit 命令を新設し、ポリシー依存関係解析部 204 がよびだされたタイミングで Commit 命令が発行されるようにする。ルータ 101 においては、Commit 命令をうけたときにステップ 1532 を実行すればよい。

#### 【 0 1 1 1 】

つづいて、図 17 を使用してポリシールール表 1113 の内容を説明する。ポリシールール表 1113 においてはルールはネットワーク・インタフェースごとにまとめて格納される。すなわち、命令先頭位置表 1701 の最初の要素がインタフェース番号 1 のネットワーク・インタフェースに関するルールのリストを指示し、命令先頭位置表 1701 の第 2 の要素がインタフェース番号 2 のネットワーク・インタフェースに関するルールのリストを指示する。しかし、図 17 においてはこれらのリストは空である。命令先頭位置表 1701 の第 3 の要素がインタフェース番号 3 のネットワーク・インタフェースに関するルールのリストを指示するが、その先頭番地は 90 (1704) である。

#### 【 0 1 1 2 】

番地 90 にはルール 1708 がおかれている。ルール 1708 は図 3 のすべてのルールの内容を融合したルールである。すなわち、図 3 における複数のルールが実行可能形式であるルール 1708 においては 1 個に融合されて表現されている。ただし、図 3 のルールがもつ情報のうちの一部はキュー設定表 1114 のなかに格納されていて、ルール 1708 のなかには存在しない。

## 【 0 1 1 3 】

ルール 1708 において、フローの始点 IP アドレスの下限 1721 には 192.168.4.1 が指定されている。フローの始点 IP アドレスの上限 1722 にも 192.168.4.1 が指定されている。したがって、ルール 1708 は始点が 192.168.4.1 であるパケットだけに作用する。始点のポート 1723 には 0 が指定されているが、これはポート番号が指定されていないことをあらわす。フローの終点 IP アドレスの下限 1724 には 0.0.0.0 が指定されている。上限 1725 には 255.255.255.255 が指定されている。これは、終点 IP アドレスが任意であることをあらわしている。終点のポート 1726 には 0 が指定されているが、これはポート番号が任意であることをあらわす。

## 【 0 1 1 4 】

DSCP 1727 にはフロー分類に DSCP をつかうときには DSCP の値を指定する。したがって、0 ～ 63 の値を指定する。しかし、ルール 1708 においてはフロー分類に DSCP をつかわないので 255 が指定されている。平均レート 1728 には帯域幅の上限を指定するときは 0 よりおおきい値を指定するが、帯域幅の上限を指定しないときは 0 を指定する。ルール 1708 においては 1000 kbps が指定されている。バーストレート 1729 には一時的な帯域幅の上限を指定するときは 0 よりおおきい値を指定するが、一時的な帯域幅の上限を指定しないときには 0 を指定する。ルール 1708 においては 0 が指定されている。

## 【 0 1 1 5 】

キュー番号 1730 においてキューを指定する。キュー番号は 0 以上、キュー数未満の値を指定する。新 DSCP 1731 には、ルールが実行されたときに DSCP をかきかえるときには 0 以上 63 以下の値を指定する。かきかえないときには 255 を指定する。ルール 1708 においては EF の DSCP すなわち 46 が指定されている。違反時動作 1732 には、平均レート 1728 またはバーストレート 1729 が指定されているときには、ルール 1708 において指定されたフローのトラフィックがこれらの帯域幅をこえたときの動作を指定する。平均レート 1728 またはバーストレート 1729 が指定されていないときには違反時動作 1732 の内容は無効である。違反時動作 1732 として指定可能な動作としては DSCP のかきかえや

パケットの廃棄などがあるが、ルール 1708 においては drop すなわち帯域幅をこえたぶんのパケットを廃棄することが指定されている。

【0116】

以上をまとめると、ルータ 101 のネットワーク・インタフェース 104 に作用するルール 1708 の意味はつぎのようになる。始点アドレスが 192.168.4.1 のフローに対して、平均レートが 1000 kbps をこえないかぎり IP パケットの DSCP に EF の DSCP すなわち 46 をマークし、キュー番号 5 のキューに入れる。1000 kbps をこえるときには、こえたぶんについてはパケットを廃棄する。ルール 1708 は以上のような意味をもっているので、後述するキューの設定 1811 とあわせると、図 3 のテンプレートに入力されたデータがあらわすすべてのルールをあわせた意味を表現している。ルール 1708 はアプリケーション・サーバ 131 からルータ 101 を経由してクライアント 141 およびクライアント 142 にいたるフローに対して適用される。

【0117】

つづいて、図 18 を使用してキュー設定表 1114 の内容を説明する。キュー設定表 1114 においてはキュー設定はネットワーク・インタフェースごとにまとめて格納される。すなわち、キュー設定表 1114 の最初の行がインタフェース番号 1 のキュー設定を指示し、キュー設定表 1114 の第 2 の行がインタフェース番号 2 のキュー設定を指示する。しかし、図 18 においてはこれらの行は空である。キュー設定表 1114 の第 3 の行がインタフェース番号 3 のキュー設定を指示するが、その先頭番地は 50 であり、スケジューリング・アルゴリズムとして PrioritySchedl すなわち優先度スケジューリングが指定されている (1804)。

【0118】

番地 50 にはインタフェース・キュー設定表 1802 がおかれている。ルータ 101 は 8 個のキューをもっている。それらには 0 から 7 までの番号がふられているので、インタフェース・キュー設定表 1802 はインデクスが 0 から 7 までの 8 行によって構成されている。また、インタフェース・キュー設定表 1802 はつぎのような欄をもっている。最低レート欄 1821 は最低保証帯域幅をあらわす。最高レート欄 1822 は最大帯域幅をあらわし、これをこえる帯域幅の入力が

あればシェイピングをおこなう。廃棄アルゴリズム欄 1824 はパケットをキューにいれる際の廃棄アルゴリズムを指定する。キューが空であればパケットが廃棄されることはないし、キューが満杯であればすべてのパケットが廃棄されるが、その中間の状態においては、廃棄アルゴリズム欄 1824 にしたがってパケットが廃棄またはキューイングされる。廃棄アルゴリズムのパラメタとして最高廃棄率 1825, 最低閾値 1826, 最高閾値 1827 が指定される。

#### 【0119】

キュー番号 5 のキューは行 1811 において設定されるが、最低レート欄 1821 には 1000, 最高レート欄 1822 には 2000, 廃棄アルゴリズム欄 1824 には WRED, 最高廃棄率 1825 には 300, 最低閾値 1826 には 50, 最高閾値 1827 には 100 が指定されている。したがって、キュー番号 5 のキューに関してはつぎのようなスケジューリングが適用される。スケジューリング・アルゴリズムとしては優先度スケジューリングが適用されるが、キュー 0 から 4 より優先度がたかく、キュー 6, 7 より優先度がひくい。最低帯域として 1000 kbps が保証されるが、最高帯域 2000 kbps をこえるとシェイピングされる。廃棄アルゴリズムとしては WRED が適用され、そのパラメタは最高廃棄率 1825 が 300, 最低閾値 1826 が 50, 最高閾値 1827 が 100 である。

#### 【0120】

つづいて、図 19 を使用してポリシールールコンパイラ 1103 の動作を説明する。ポリシールールコンパイラ 1103 の実行が開始されると、わたされたすべてのルールについてステップ 1901 から 1921 までの処理をくりかえす。まず、ステップ 1901 において、わたされたルールのなかで未処理のルールがあるかどうかを判定し、まだあればステップ 1902 にすすみ、もうなければポリシーコンパイラ 1103 の処理を終了する。ステップ 1902 においては、当該のルールを変換したコードがすでにポリシールール表 1113 にふくまれていれば、それを削除するか、または無効な命令によって置換する。

#### 【0121】

そして、ステップ 1903 において、当該のルールの型を判定する。それがフロー分類型 (Classification) であれば分類命令生成処理 1911 を実行し、契約違



反管理型 (Policing) であれば警戒命令生成処理 1912 を実行し, QoS 動作型 (QoSAction) であればQoS 動作命令生成処理 1913 を実行し, スケジューリング型 (Scheduling) であればスケジューリング設定生成処理 1914 を実行する。分類命令生成処理 1911, 警戒命令生成処理 1912, QoS 動作命令生成処理 1913 の実行後はステップ 1921 にすすみ, スケジューリング設定生成処理 1914 の実行後はステップ 1901 にもどってつぎのルールの処理にうつる。

#### 【0 1 2 2】

ステップ 1921 においてはポリシーソースルール DB における当該のルールのコード欄 1317 に, 生成した命令の先頭アドレスを記入する。

#### 【0 1 2 3】

つづいて, 図 20 を使用して分類命令生成処理 1911 について説明する。分類命令生成処理 1911 の実行が開始されると, まずステップ 2001 において指定インタフェースに命令が格納できる場所をわりあてる。つづいて, ステップ 2002 において, 当該の命令が当該インタフェースへの最初の命令なら, その先頭アドレスを命令先頭位置表 1701 のにおける当該インタフェースに対応する要素 1704 に記入する。さらに, ステップ 2003 において, 図 13 における当該のルールに関する部分から, 生成した命令のなかに, フローの始点 IP アドレス範囲とポート, 終点 IP アドレス範囲とポートおよび DSCP を当該の命令に記入する。

#### 【0 1 2 4】

つづいて, 図 21 を使用して警戒命令生成処理 1912 について説明する。警戒命令生成処理 1912 の実行が開始されると, まずステップ 2101 において当該のルールが参照する変数を定義するルールを変換した結果の命令を特定する。すなわち, ルール識別子が #2 の命令についていえば, まずポリシーソースルール DB 1111 の条件欄が指示しているテーブル (B) (431) を参照することによって, 当該ルールが番号 1 の変数を参照していることがわかるので, 変数定義表 1401 の 1 行めを参照することによってそれを定義しているのがルール識別子が #1 のルールであることがわかる。つぎに, ポリシーソースルール DB 1111 のルール識別子が #1 の命令のコード欄 1317 を参照することによって, 命令の先頭アドレスが 90 であることがわかり, 命令を特定することができる。

【 0 1 2 5 】

つぎに、ステップ 2102 において当該の命令の最高レート欄 1728 およびバーストレート欄 1729 にポリシーソースルール DB 1111 の条件欄が指示しているテーブル (B) における最高レート欄 432 およびバーストレート欄 433 の値をコピーする。

【 0 1 2 6 】

つづいて、図 22 を使用して QoS 動作命令生成処理 1913 について説明する。QoS 動作命令生成処理 1913 の実行が開始されると、まずステップ 2201 において、当該のルールが参照する変数を定義するルールの変換結果である命令を特定する。その具体的な方法はステップ 2101 とおなじである。そして、前記の命令の新 DSCP 欄 (NewDSCP) 1731 に当該ルールの動作部がふくむ DSCP の値を記入する。ルール識別子が #3 のルールについては、テーブル (C) における DSCP 欄 441 にふくまれる 46 を記入する。

【 0 1 2 7 】

つぎに、ステップ 1933において、前記の命令が使用するキューがわりあてずみであるかどうかを判定する。わりあてずみならばステップ 1951 にすすみ、そうでなければステップ 1941 にすすむ。ステップ 1941 においてはポリシールールソース DB 1111 における当該ルールが参照するスケジューリングルールのコード欄 1317 に、現在わりあてられていないキューの番号を記入する。つぎに、ステップ 1942 において、前記のスケジューリング・ルールで指定された上位スケジューリングルールにおいて指定されたスケジューリング・アルゴリズムと、最低レート、最高レートをキュー設定表 1114 に記入する。

【 0 1 2 8 】

ルール識別子が #3 のルールについては、テーブル (E) におけるスケジューリング・アルゴリズム 463 すなわち優先度スケジューリング (PrioritySched1) をキュー設定表 1114 のインタフェース番号 3 の行のスケジューリング・アルゴリズム欄 1802 に記入し、最低レート 1000 (461)、最高レート 2000 (462) をインタフェース・キュー設定表 1808 のキュー番号 5 の行 1811 の最低レート欄 1821 および最高レート欄 1822 に記入する。そして、ステップ 1951 にす

すむ。

【0129】

ステップ 1951 においては、ポリシールールソース DB における当該ルールが参照するスケジューリングルールのコード欄 1317 によって指定されているキューに対応するインタフェース・キュー設定表 1808 のキュー番号 5 の行 1811 の廃棄アルゴリズム欄 1824, 最高廃棄率欄 1825, 最低閾値欄 1826, 最高閾値欄 1827 の値を記入する。

【0130】

つづいて、図 23 を使用してスケジューリング設定生成処理 1914 について説明する。スケジューリング設定生成処理 1914 においては、ステップ 2301 において、ポリシールールソース DB における当該ルールのコード欄 1317 によって指定されているキューに対応するインタフェース・キュー設定表 1808 のキュー番号 5 の行 1811 に、当該ルールにおける最低レート 461, 最高レート 462 の値を記入する。

【0131】

以上で基本の実施例の説明をおわる。以下、前記の実施例の一部を変更した実施例について説明する。

【0132】

第 1 に、前記の基本の実施例においては、ポリシールール依存関係解析部 204 において変数の参照関係によって生じる依存関係だけを解析している。このような依存関係を下記の文献においてはフロー依存関係とよんでいる。

【0133】

金田 泰 他著、配列の大域データフロー解析法、情報処理学会論文誌、第 28 巻第 6 号、1987 年、567 ページから 576 ページ。

【0134】

J. R. Allen 他著、Conversion of Control Dependence to Data Dependence、国際学会会議録 The 10th Annual ACM Symposium on Principles of Programming Languages、1983 年、177 ページから 189 ページ。

上記の文献においては、フロー依存関係のほかに、出力依存関係、逆依存関係

という 2 種類のデータ依存関係と、制御依存関係とを定義している。上記に文献において定義されているデータ依存関係および制御依存関係は手続き型言語で記述されたプログラムに対して定義されているが、ルール型言語に対しても同様に定義することができる。また、これらの定義にしたがって複数のルール間のデータ依存関係および制御依存関係の有無を判定することができる。これらの依存関係の有無がわかれば、ポリシールール依存関係解析部 204 においてそれを解析し、その関係にもとづくルールの推移閉包をもとめることができる。

#### 【0135】

また、ポリシー整合性検査部 203 における条件が排他であるかどうかの判定をポリシールール依存関係解析部 204 においてあわせておこない、排他でないときはルール間に依存があると判定するようにすれば、条件の非排他性にもとづく依存もあわせて解析することができる。前記の基本の実施例においてはポリシールール依存関係解析部 204 において出力依存関係、逆依存関係、制御依存関係を解析する必要はないが、ポリシーサーバ 103 とルータ 101 間のインタフェース仕様によっては、ルールの追加・削除・更新の際にこれらの解析が必要になる。

#### 【0136】

第 2 に、前記の基本の実施例においては、ポリシーサーバ 103 からルータ 101 に送信する命令として、Deploy, Undeploy, Redeploy の 3 種類を使用していた。しかし、つぎのような構成も可能である。命令の種類は Load, Deploy2, Undeploy2, Unload の 4 種類とする。Load 命令の形式は Deploy 命令の形式 2401 とひとしく、Deploy2 命令, Undeploy2 命令, Unload 命令の形式は Redeploy 命令の形式 2431 とひとしい。ただし、Load 命令および Unload 命令はインタフェース番号 2404 を指定しない。すなわち、インタフェース番号の欄 2404 は空とする。

#### 【0137】

ポリシーサーバ 103 は Load 命令によってルールの内容をルータ 101 に送信する。また、ルータ 101 に格納されているルール識別子 2433 によって指定されたルールを Deploy2 命令によって特定のインタフェースに作用させることを

指示する。したがって、前記の実施例における Deploy 命令の機能は、Load 命令と Deploy2 命令のくみあわせによって実現される。また、Undeploy2 命令によって特定のインタフェースに作用しているルール識別子 2433 によって指定されたルールをそのインタフェースに作用しないようにすることを指示する。Undeploy2 命令が発行されても、ルータ 101 からルールが削除されることはない。さらに、Unload 命令によってルータ 101 からルール識別子 2433 によって指定されたルールを削除する。

## 【0 1 3 8】

この構成をとることにより、ルータ 101 に対して明示的にルールの削除を指示することができるため、ルータ 101 の資源をより効率的に管理することができる。また、ルータ 101 が独自の判断で削除したために Redeploy 命令を適用することができなくなったルールに対してポリシーサーバ 103 が Redeploy 命令を発行するというあやまりを防止することができる。

## 【0 1 3 9】

第 3 に、前記の基本の実施例においては複数のルールをルータ 101 に転送する際には、それらがわずかな部分をのぞいて同一であっても、複数のルール全体を転送する必要があった。ラベル変数値だけがことなるルールを下記の Deploy2 命令を新設することによって、このようなばあいの転送量を検証させることができる。図 26 を使用して Deploy2 命令について説明する。Deploy2 命令 2601 はすでに送信先のルータに格納されているルールを複写して、それが定義または使用するラベル変数の値だけを置換して、あらたなルールとして格納することを指示する命令である。Op コード 2602 は Deploy2 という値をふくんでいるが、これはこのデータが Deploy2 命令をあらわしていることをしめす。ルール識別子欄 2603 は当該 Deploy 命令がふくむルールのルール識別子をあらわす。インタフェース欄 2604 は当該 Deploy 命令が作用すべきネットワーク・インタフェースの番号をあらわす。旧ルール識別子欄 2605 は複写元のルールのルール識別子をあらわす。新ラベル欄 2606 は置換後のラベル変数の値をあらわす。

## 【0 1 4 0】

Deploy2 命令 2601 においては変更すべきラベル変数の値は 1 個だけだが、

複写元のルールにおいて複数のラベル変数を定義または使用していて、両方の値を置換するばあいには、さらに欄の数をふやせばよい。また、ラベル変数の値以外の値を置換するばあいも同様である。

## 【0 1 4 1】

Deploy2 命令 2601 を使用することによって、Deploy 命令 2401 を使用するばあいに比べてポリシーサーバ 103 からルータ 101 へのデータ転送量を減少させることができる。ポリシーサーバ 103 とルータ 101 とのインタフェース仕様においてマクロなルールを使用するばあいは類似したルールが使用される可能性は比較的ひくい、この実施例におけるように細分化されたルールを使用するばあいには類似したルールが多用され、したがって Deploy2 命令を使用することによる効果はおおきいとかんがえられる。

## 【0 1 4 2】

第 4 に、前記の基本の実施例においては、ポリシーサーバ 103 が送信する命令をルータ 101 が直接解釈することができることを前提にしていた。しかし、既存のルータをポリシーサーバに接続するばあいには、これはかならずしも可能でない。このようなばあいには、図 11 におけるルータ 101 のかわりに、図 25 におけるプロキシー 2501 とルータ 2502 を使用すればよい。この構成においては、図 11 の構成と以下の部分だけがことになっている。プロキシー 2501 においては、キュー設定表 2516 とポリシールール表 2517 を主記憶またはハードディスクにおくが、その内容はキュー設定表 1114 およびポリシールール表 1113 とひとしい。コマンド送信部 2511 はキュー設定表 2516 とポリシールール表 2517 の内容をルータ 2502 に送信する。ルータ 2502 においては、プロキシー 2501 からの受信データをキュー設定表 1114 とポリシールール表 1113 に格納する。この構成をとることにより、ポリシーサーバ 103 が送信する命令を解釈することができないルータ、とくにすでに配備されたルータに対して本発明を適用することができる。

## 【0 1 4 3】

第 5 に、前記の基本の実施例においては、ポリシールールコンパイラ 1103 において複数のルールを融合して 1 個の実行可能な形式のルールを生成したが

、ルール of 分解をおこなうことはなかった。これは、ポリシーサーバ 103 とルータ 101 と of インタフェース仕様において、ルールがすでに十分に細分化されているためである。しかし、前記 of インタフェース仕様においてマクロなルールを使用することがきめられていて、しかもルータにおける実行可能なルール of 形式がより細分化されたものであるばあいには、ポリシールールコンパイラ 1103 においてルールを分解することが必要になる。たとえば、前記 of インタフェース仕様においてきめられたルール of 形式が 1708 であり、実行可能な形式が 401 であるとする。このときはラベル変数を導入し、その値として 1, 2, 3 を使用することによって、ルールを分解することができる。

#### 【0 1 4 4】

ルール of 分解はつぎのようなばあいにも適用することができる。オペレータは 1708 のような形式 of ルールを入力するが前記 of インタフェース仕様においては前記 of 実施例のよう to 細分化されたルールを使用するばあいには、ポリシーサーバ 103 のポリシールール依存関係解析部 204 とポリシースケジューリング部 205 と of あいだにルール of 分解をおこなうプログラムを挿入すればよい。

#### 【0 1 4 5】

第 6 に、前記 of 基本 of 実施例におけるポリシーコンパイラ 1103 においては、ルールを更新する際にはまずそのルールに対応する命令を削除してからあらたな命令を生成する。しかし、この方法ではルールが一時的に作用しなくなる。この作用 of 中断をさけるためには、つぎのような方法をとればよい。

#### 【0 1 4 6】

まず、1 個 of ルールが更新されるあいだだけもとのルール of 作用を継続すればよいのであれば、ステップ 1902 をスキップし（すなわちステップ 1901 の条件がなりたったときにはステップ 1903 にすすむようにし）、ステップ 1911 から 1914 の命令生成時には生成した命令が作用しないようにしておく。そして、ステップ 1921 および 1914 の終了後に前記 of 命令を発効させる。既存 of 命令を置換するばあいには、このときに既存 of 命令 of 無効化と前記 of 命令 of 発効とを同時にこなう。これによって作用が中断するのをさけることができる。

## 【0 1 4 7】

また、複数のルールを同時に更新したいのであれば、つぎのようにすればよい。ポリシースケジューリング表 213 の項目として switch 項目を新設する。switch 項目に対しては deploy 項目と同一の次時刻 723 と時刻指定 724 を記入しておく。ポリシースケジューリング部 205 のステップ 803 において入力した項目の deploy 項目をすべて生成した直後に switch 項目を生成する。ポリシーサーバ 103 からルータ 101 に送信する命令の一種として Switch 命令を新設する。Switch 命令の形式は Redeploy 命令の形式 2431 とひとしいが、ルール識別子 2433 は指定しない。ポリシー送信部 206 においてはステップ 1004 において switch イベントが検出したときに指定ルータに対してネットワーク・インタフェース番号を指定して Switch 命令を送信する。

## 【0 1 4 8】

ルータ 101 においては Switch 命令を受信するまではポリシールールコンパイラ 1103 において生成した命令は有効にならないものとする。ポリシー受信部 1101 においては、ステップ 1502 において Switch 命令を検出すると、指定されたネットワーク・インタフェースに対して、ポリシールールコンパイラ 1103 において生成した命令を有効にし、置換すべき既存の命令があるときにはそれを同時に無効にするように指示する。このようにあらかじめ生成した命令を一括して追加または置換するためには、命令用の記憶装置の一部または全部を 2 面化して、指示された時点で 2 個の面をいれかえればよい。

## 【0 1 4 9】

第 7 に、前記の基本の実施例においては、オペレータが各ルールを個別に入力しているが、細分化されたルールを入力するのはかならずしも容易でない。より容易に入力できるようにするには、つぎのようにすればよい。DiffServ における標準的なサービスに対してはルールのくみあわせをテンプレートのかたちで表現して、テンプレート内のパラメタだけをうめれば特定のフローに対するサービスが定義できるようにすればよい。図 27 を使用してこの実施例について説明する。



## 【 0 1 5 0 】

テンプレート 2701 は契約違反管理をしない単純なサービスのためのテンプレートである。テンプレート 2701 はフロー分類ルール 2711, QoS 動作ルール 2712, スケジューリング・ルール 2713 によって構成され、矢印 2714 および矢印 2715 によってそれらが結合されている。テンプレート 2701 はテンプレート 301, テンプレート 321, テンプレート 341 をくみあわせたテンプレートである。矢印 2714 はフロー分類ルール 2711 と QoS 動作ルール 2712 とをつなぐラベル変数値をあらわす。矢印 2714 は、テンプレート 301, テンプレート 321 におけるラベル欄 307 および 323 に相当する。また、矢印 2715 は、テンプレート 321, テンプレート 341 におけるラベル欄 327 および 343 に相当する。

## 【 0 1 5 1 】

テンプレート 2702 は契約帯域をこえるパケットを廃棄するサービスのためのテンプレートである。テンプレート 2702 はフロー分類ルール 2721, 契約違反管理ルール 2722, QoS 動作ルール 2723, スケジューリング・ルール 2724, QoS 動作ルール 2725 によって構成され、矢印 2726, 矢印 2727, 矢印 2728 および矢印 2729 によってそれらが結合されている。QoS 動作ルール 2725 においては動作の既定値として drop すなわち廃棄が指定されている。

## 【 0 1 5 2 】

なお、以上の実施例はポリシーによる QoS の制御に関する実施例だったが、この発明の方法はポリシーサーバからルータ等のネットワーク・ノードに対して他の機能をもつルールをダウンロードする際にも適用される。たとえば、スイッチングやルーティングを制御するルール、NAT (Network Address Translation) のようにパケットがふくむフローの始点や終点に関する情報やペイロードにふくまれるアドレスを変換するルール、ペイロードにふくまれる情報をもとに計算をおこない、その結果をペイロードにかきこむルール、さらには複数のパケットに作用して、それらのペイロードがふくむ情報を入力してあらたなパケットを生成するルールなどに対しても適用することができる。

## 【 0 1 5 3 】

## 【発明の効果】

本発明のネットワーク制御方法を使用すれば、ポリシールール間のデータの依存性を解析する手段を使用することにより、ルータにおいてポリシールールを実行可能な形式に変換する際に変換すべき最小限のポリシールールやデータのセットをもとめることができる。

## 【 0 1 5 4 】

また、ポリシーサーバからルータに対してポリシールールを指定する際には、当該ポリシールールがルータに格納されているかどうかを判定する手段を使用することにより、当該ポリシールールの内容を転送せずにその識別子だけを転送することができる。これにより、転送すべきデータ量を最小限にすることができる。したがって、ネットワークの混雑を最小限におさえるとともに、ダウンロード時間およびポリシールールの変換にかかる時間を最小限にし、ポリシー制御が中断することがないかまたは中断時間が最小限ですむようにし、またルータに過大な負荷がかかることがないようにすることができる。

## 【図面の簡単な説明】

## 【図 1】

本実施例におけるネットワーク構成図である。

## 【図 2】

図 1 におけるポリシーサーバの構成図である。

## 【図 3】

ポリシールールをオペレータが入力するためのテンプレートとそこに入力されたデータをしめす図である。

## 【図 4】

図 2 のポリシーリポジトリの内容をしめす図である。

## 【図 5】

図 2 のポリシー入力処理部の処理フローをしめす図である。

## 【図 6】

図 2 のポリシー整合性検査部の処理フローをしめす図である。

【図 7】

図 2 におけるポリシースケジュール表の内容をしめす図である。

【図 8】

図 2 のポリシースケジューリング部の処理フローをしめす図である。

【図 9】

図 2 のネットワーク構成管理表の内容をしめす図である。

【図 1 0】

図 2 のポリシー送信部の処理フローをしめす図である。

【図 1 1】

図 1 におけるルータの構成図である。

【図 1 2】

図 11 におけるネットワーク・インタフェースの構成図である。

【図 1 3】

図 11 におけるポリシーソースルール DB の内容をしめす図である。

【図 1 4】

図 1 におけるデータ参照表の内容およびそのグラフによる表現をしめす図である。

【図 1 5】

図 11 のポリシー受信部の処理フローをしめす図である。

【図 1 6】

図 1 のポリシールール依存関係解析部の処理フローをしめす図である。

【図 1 7】

図 11 におけるポリシールール表の内容をしめす図である。

【図 1 8】

図 11 におけるキュー設定表の内容をしめす図である。

【図 1 9】

図 11 のポリシーコンパイラの処理フローをしめす図である。

【図 2 0】

図 19 の分類命令生成処理のフローをしめす図である。

【図 2 1】

図 19 の警戒命令生成処理のフローをしめす図である。

【図 2 2】

図 19 の QoS 動作命令生成処理のフローをしめす図である。

【図 2 3】

図 19 のスケジューリング設定生成処理のフローをしめす図である。

【図 2 4】

図 1 におけるポリシーサーバー-ルータ間の通信データの形式をしめす図である

【図 2 5】

他のルータ構成のしめす図である。

【図 2 6】

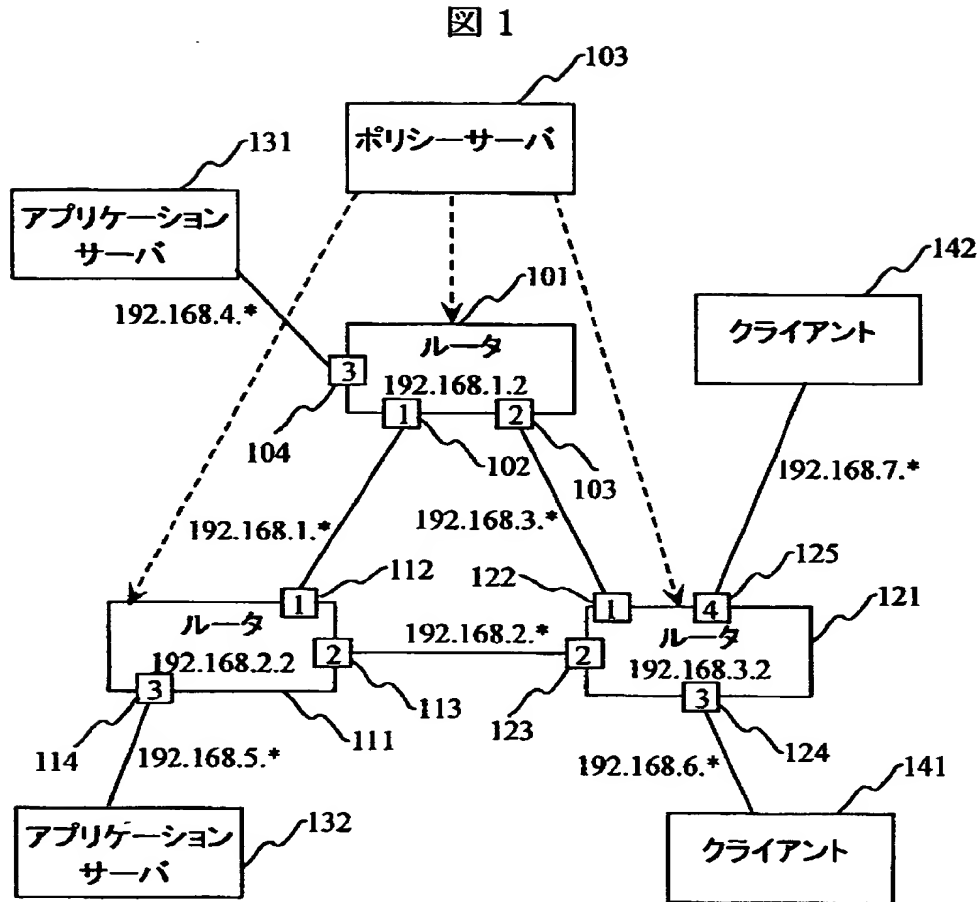
図 24 の通信データ形式に追加するべき通信データ形式をしめす図である。

【図 2 7】

ポリシールールをオペレータが入力するためのテンプレート。

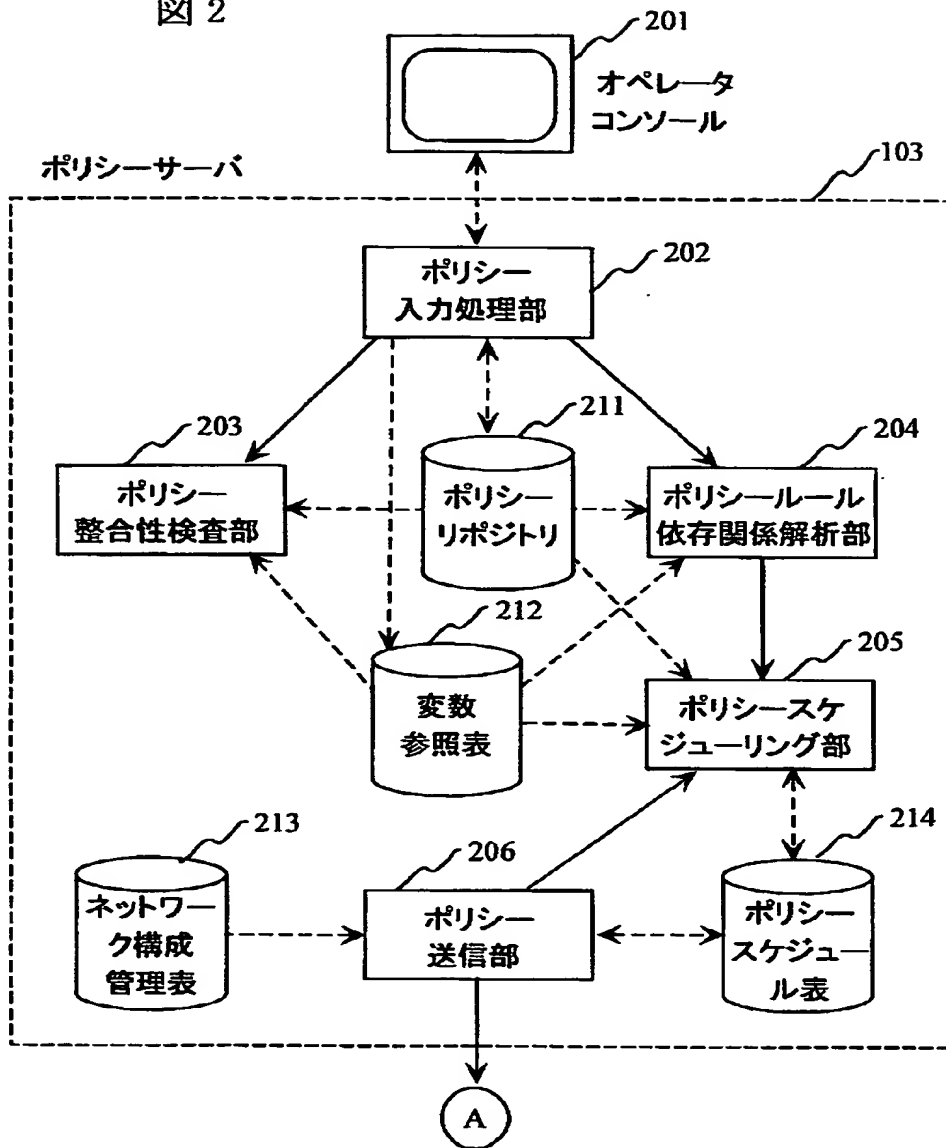
【書類名】 図面

【図 1】



【図 2】

図 2



【図 3】

図 3 (a)

301  
↓

Rule type	Condition				Action	Time	
Classification	Protocol	Source IP	Destination IP	DSCP	Labeling	From	To
	TCP	192.168.4.1	-	-	VideoSource	Sat	Sun

↑302    ↑303    ↑304    ↑305    ↑306    ↑307    ↑308    ↑309

321  
↓

Rule type	Condition				Action	Time	
Policing	Label	Rate unit	Committed rate	Burst rate	Labeling	From	To
	VideoSource	kbps	>= 1000	-	VideoPolice	9:00	17:00

↑322    ↑323    ↑324    ↑325    ↑326    ↑327    ↑328    ↑329

341  
↓

Rule type	Condition	Action						Time		
QoSAction	Label	DS	Discard				Labeling	From	To	
		CP	alg	ratio	unit	min	max			
	VideoSource	EF	WRED	300	packets	50	100	VideoSchedl	Sat	Sun

↑342    ↑343    ↑344    ↑345    ↑346    ↑347    ↑348    ↑349    ↑350    ↑351    ↑352

(b)

361  
↓

Rule type	Condition	Action							Time	
QoSAction	Label	DS	Discard					Labeling	From	To
	CP	alg	ratio	unit	min	max				
	VideoPolice	-	DropAll	-	-	-	-	-	9:00	17:00

↑

362

↑

363

↑

364

↑

365

↑

366

↑

367

↑

368

↑

369

↑

370

↑

371

↑

372

381  
↓

Rule type	Condition	Action				Time	
Scheduling	Label	Rate unit	Min rate	Max rate	Parent	From	To
	VideoSchedl	kbps	1000	2000	PrioritySchedl	Sat	Sun

↑

382

↑

383

↑

384

↑

385

↑

386

↑

387

↑

388

↑

389

【図 4】

図 4 (a)

401  
↓

Rule ID	Rule type	Condition	Action	
#1	Classification	→ (A)	1	← 402
#2	Policing	→ (B)	2	← 403
#3	QoSAction	1	→ (C)	← 404
#4	QoSAction	2	→ (D)	← 405
#5	Scheduling	3	→ (E)	← 406

↑            ↑            ↑            ↑  
411            412            413            414

(b)

(A) →	Protocol	TCP	← 421
	SourceIPmin	192.168.4.1	← 422
	SourceIPmax	192.168.4.1	← 423
	SourcePort	null	← 424
	DestinationIPmin	null	← 425
	DestinationIPmax	null	← 426
	DestinationPort	null	← 427
	DSCP	null	← 428

(B) →	Label	1	← 431
	CommittedRate	1000	← 432
	BurstRate	null	← 433

(C) →	DSCP	46	← 441
	DiscardAlgorithm	WRED	← 442
	DiscardRatio	300	← 443
	ThresholdUnit	Packets	← 444
	MinThreshold	50	← 445
	MaxThreshold	100	← 446
	Labeling	3	← 447

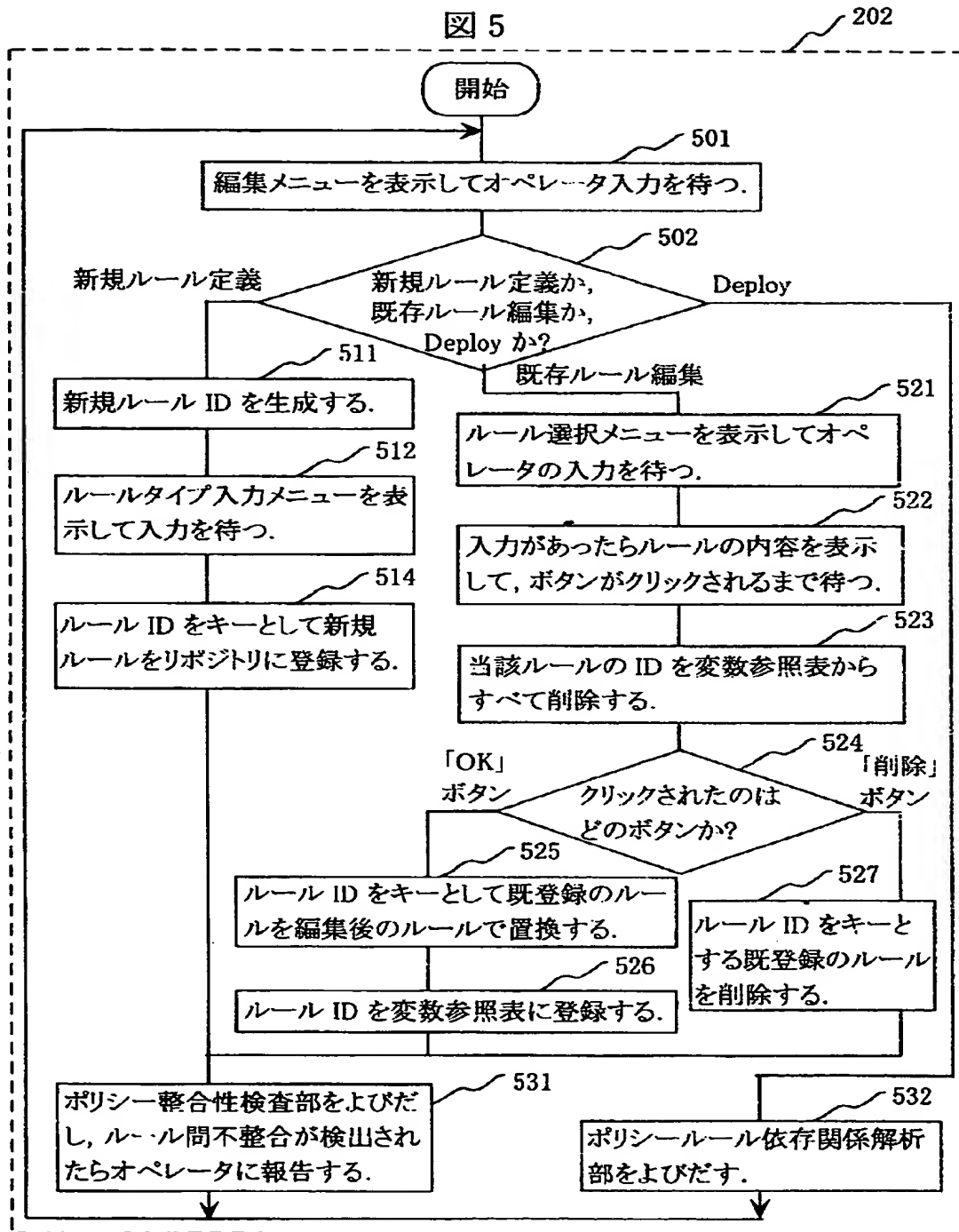
(D) →	DSCP	255	← 451
	DiscardAlgorithm	DropAll	← 452
	DiscardRatio	null	← 453
	ThresholdUnit	null	← 454
	MinThreshold	null	← 455
	MaxThreshold	null	← 456
	Labeling	null	← 457

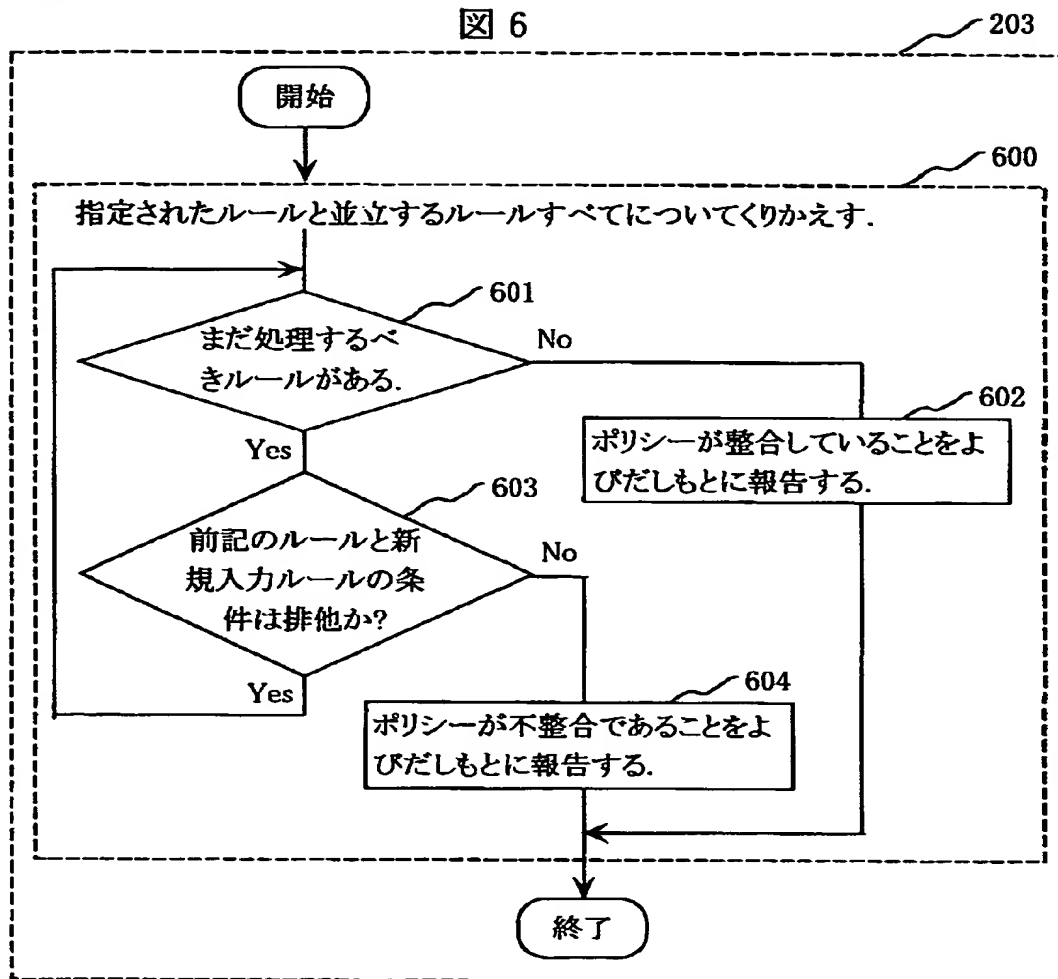
(E) →	MinRate	1000	← 461
	MaxRate	2000	← 462
	Parent	PrioritySchedl	← 463



【図 5】



【図 6】



【図 7】

図 7

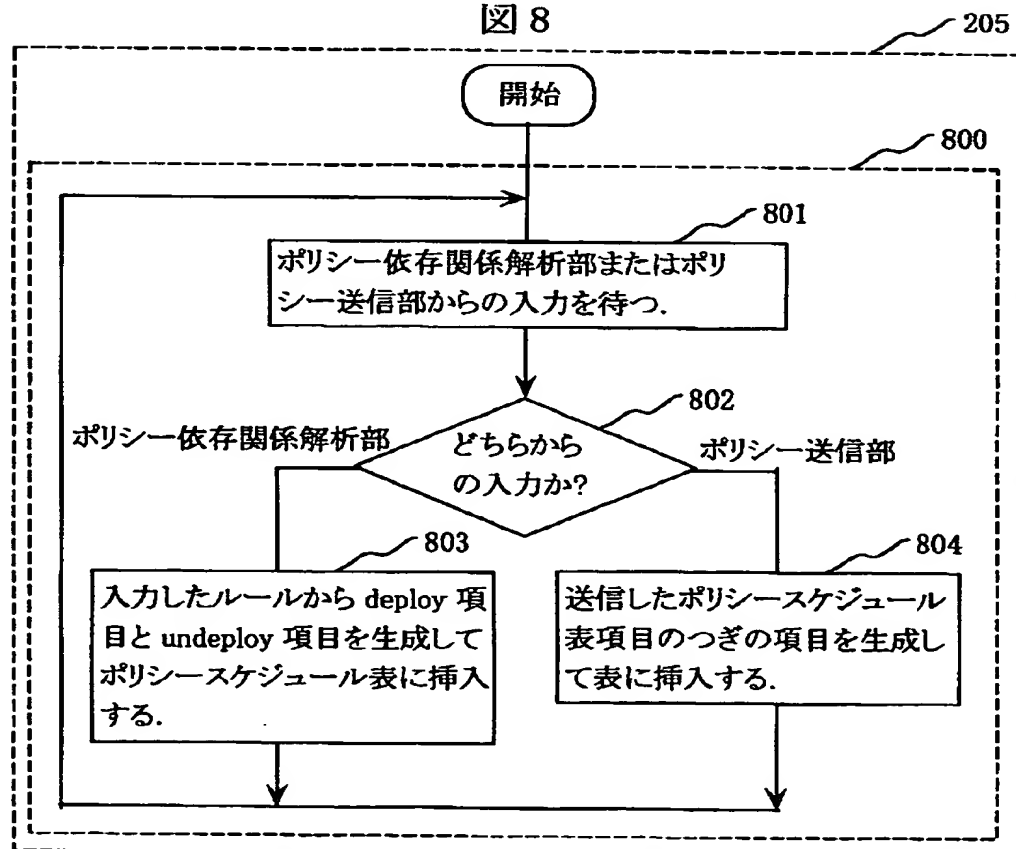
213  
↓

Rule ID	Event	Next time	Time	
#1	Deploy	1999-11-27 0:00	Sat	← 702
#3	Deploy	1999-11-27 0:00	Sat	← 703
#5	Deploy	1999-11-27 0:00	Sat	← 704
#2	Deploy	1999-11-27 9:00	9:00	← 705
#4	Deploy	1999-11-27 9:00	9:00	← 706
#2	Undeploy	1999-11-27 17:00	17:00	← 707
#4	Undeploy	1999-11-27 17:00	17:00	← 708
#1	Undeploy	1999-11-29 0:00	Sun	← 709
#3	Undeploy	1999-11-29 0:00	Sun	← 710
#5	Undeploy	1999-11-29 0:00	Sun	← 711

↑      ↑      ↑      ↑  
721    722    723    724

【図 8】

図 8



【図 9】

図 9

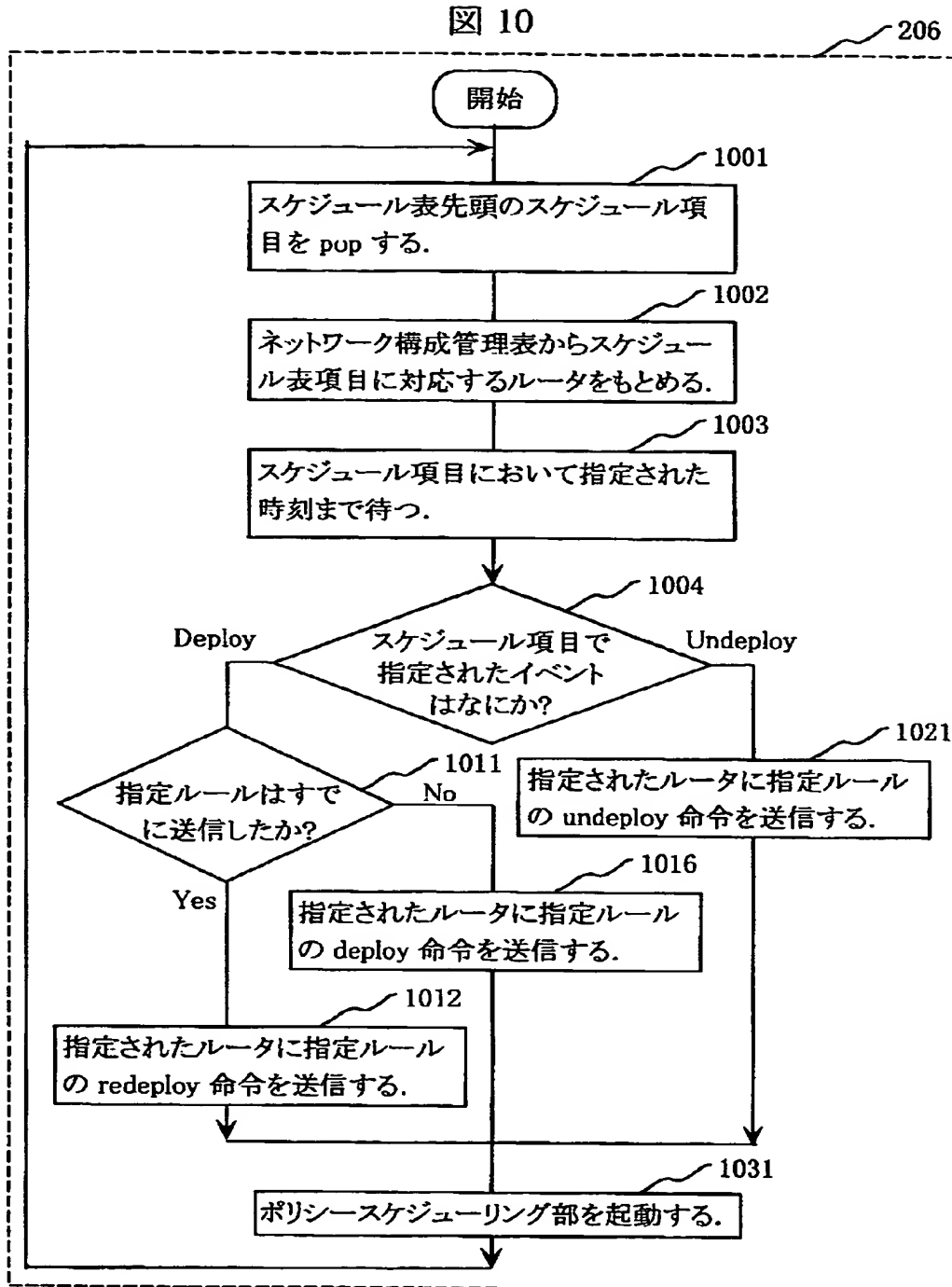
212  
↓

Target IP	Router IP	Interface	
192.168.4.*	192.168.1.2	3	← 902
192.168.5.*	192.168.2.2	3	← 903
192.168.6.*	192.168.3.2	3	← 904
192.168.7.*	192.168.3.2	4	← 905

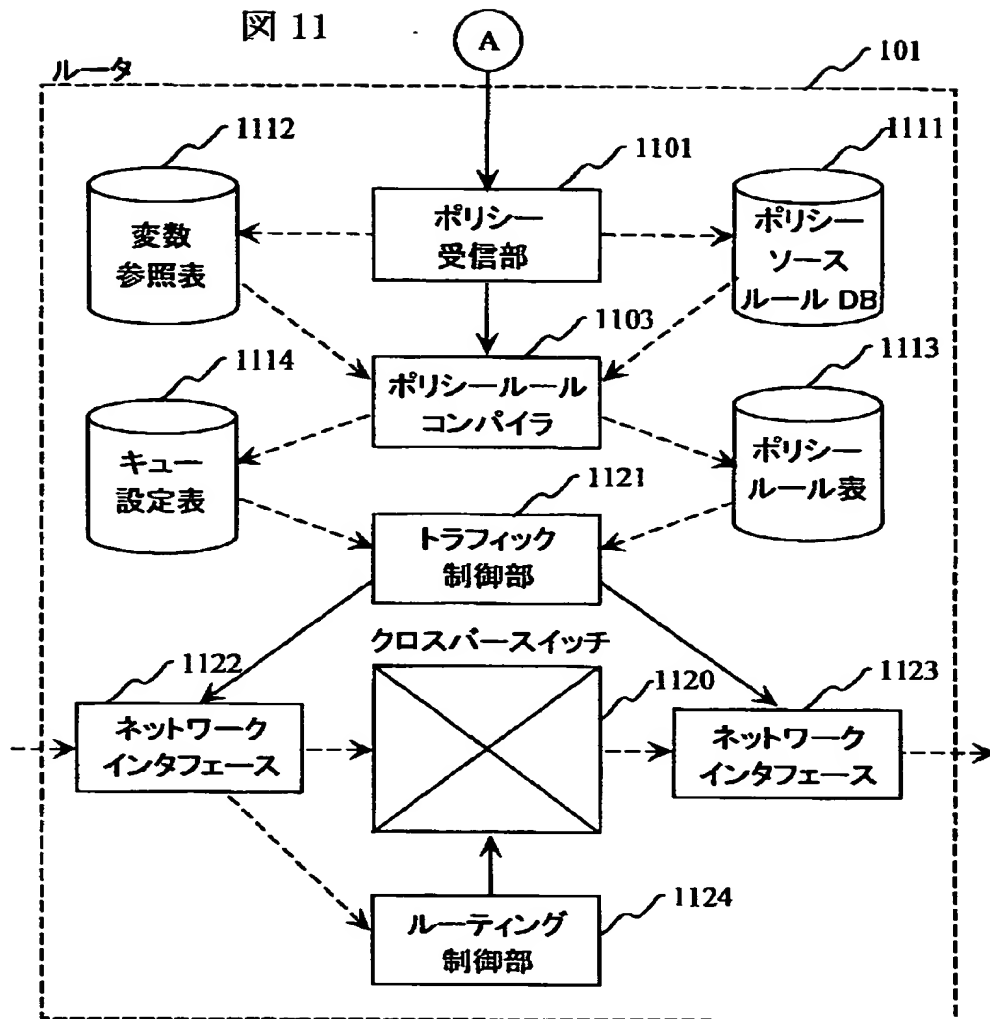
↑                    ↑                    ↑

911                    912                    913

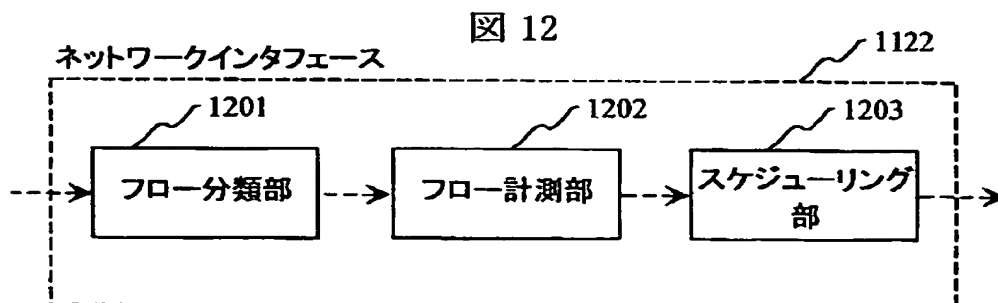
【図 1 0】



【図 1 1】



【図 1 2】



【図 1 3】

図 13

1111  
↓

Rule ID	Rule type	Condition	Action	Interface	Codep	
#1	Classification	→ (A)	1	3	90	← 1302
#2	Policing	→ (B)	2	3	90	← 1303
#3	QoSAction	1	→ (C)	3	90	← 1304
#4	QoSAction	2	→ (D)	3	90	← 1305
#5	Scheduling	3	→ (E)	3	5	← 1306

↑            ↑            ↑            ↑            ↑            ↑

1311        1312        1313        1314        1316        1317

【図 1 4】

図 14 (a)

1401    変数定義表

↓

1	#1	← 1411
2	#2	← 1412
3	#3	← 1413

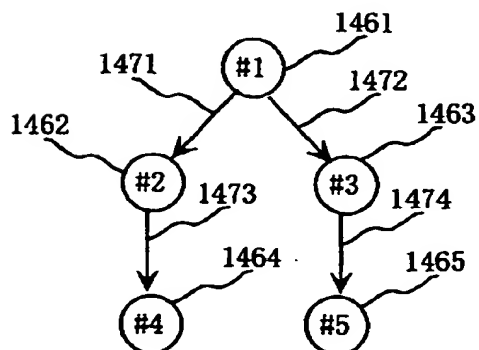
  

1421    1422    変数使用表

↓            ↓

1	#2	#3	← 1431
2	#4		← 1432
3	#5		← 1433

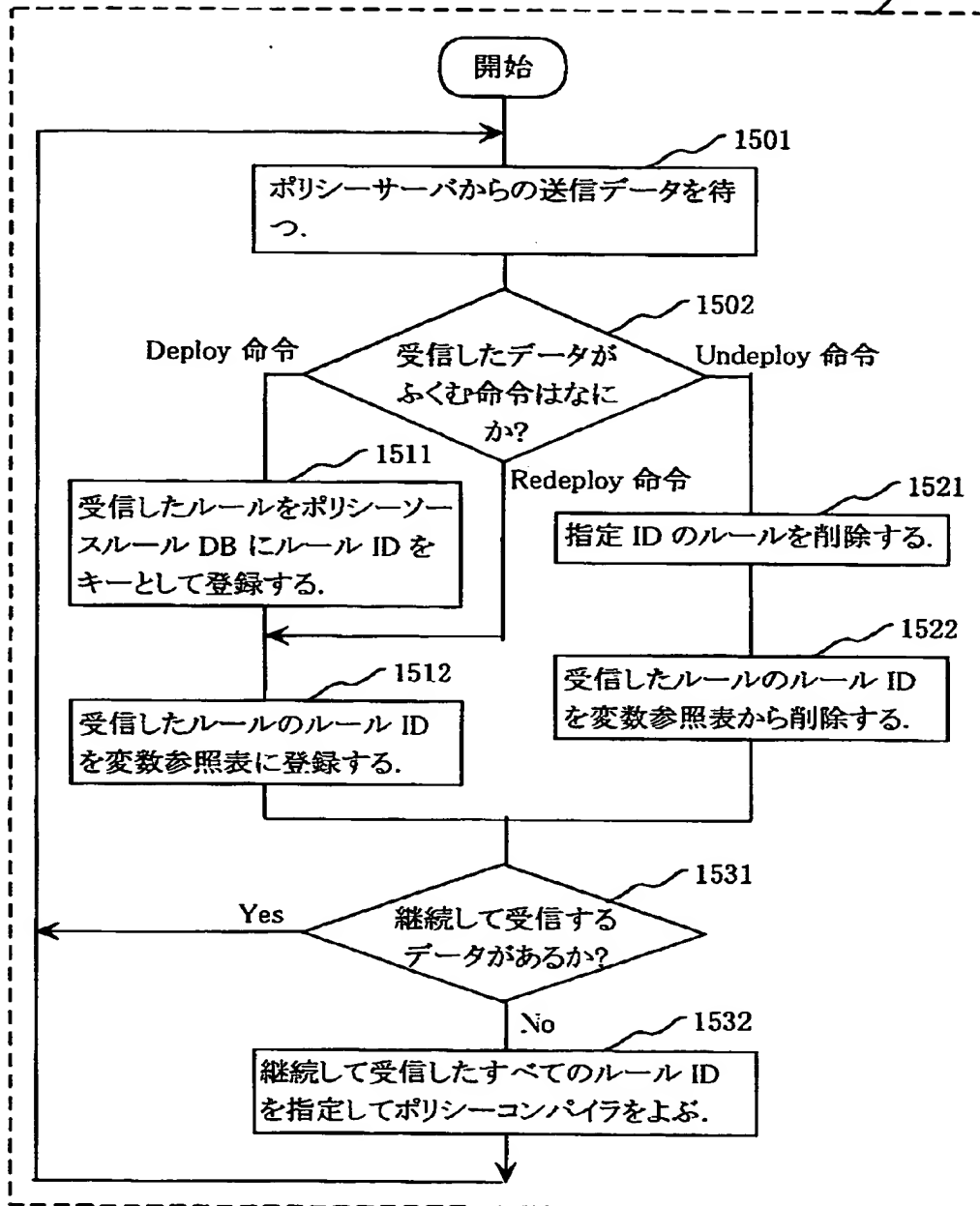
(b)



【図 1 5】

図 15

1101

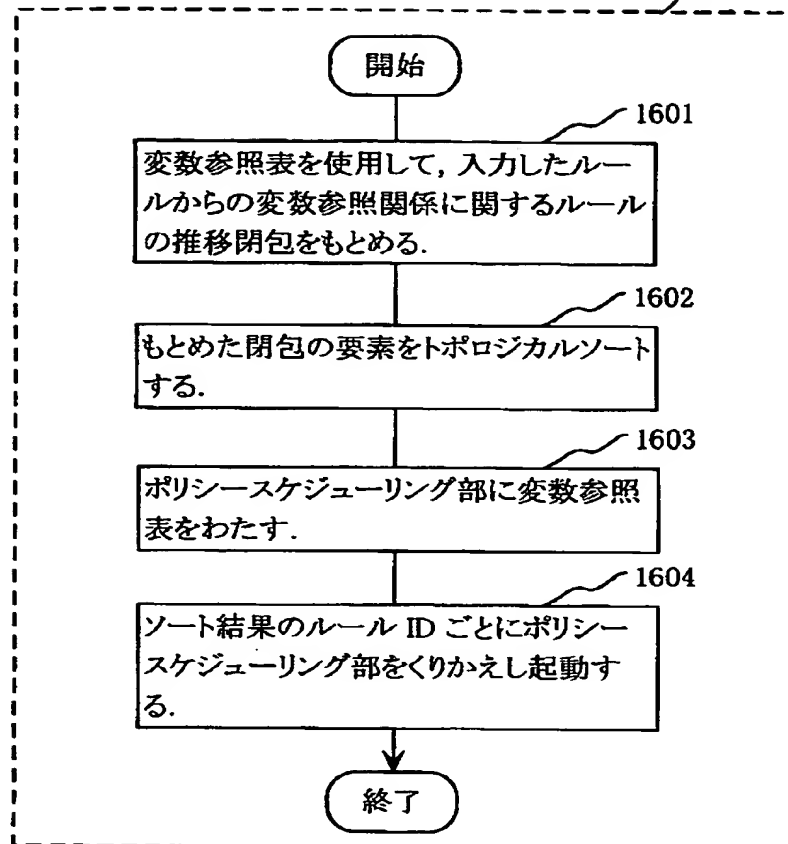




【図 1 6】

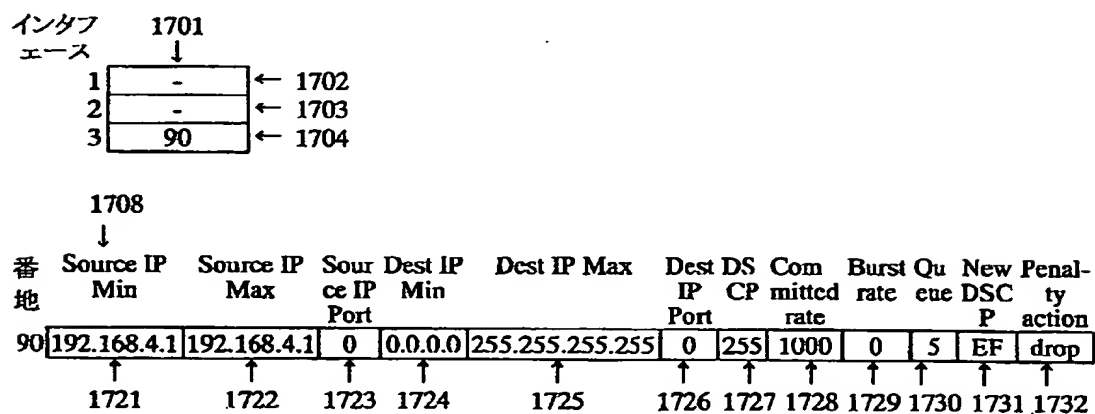
図 16

204



【図 1 7】

図 17



【図 1 8】

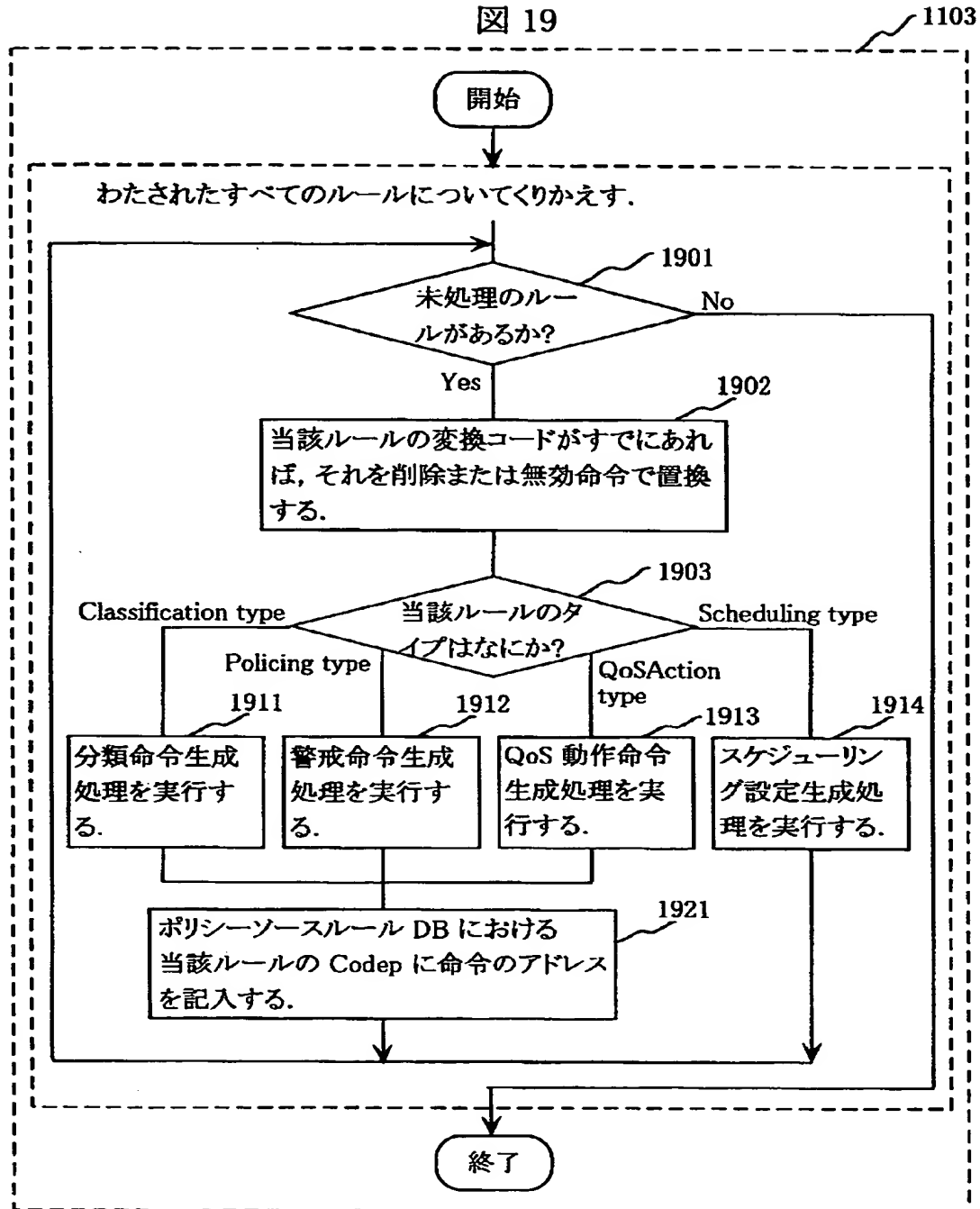
図 18

1114 インタフェース		1801 ↓ 番地	1802 ↓ Schedl alg			
1	-	-	-	← 1803		
2	-	-	-	← 1804		
3	50	PrioritySchedl		← 1805		

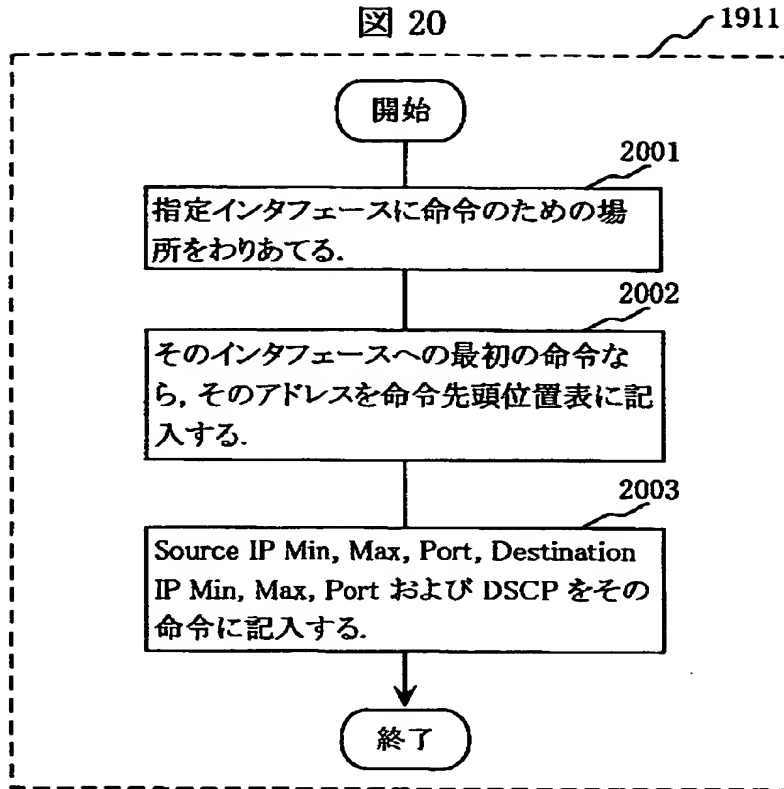
  

1808 ↓ 番地 インデックス		MinRate	MinRate	Discard alg	Discard ratio	Discard min	Discard max		
50	0								
	1								
	2								
	3								
	4								
	5	1000	2000	WRED	300	50	100	← 1811	
	6								
	7								
		↑ 1821	↑ 1822	↑ 1824	↑ 1825	↑ 1826	↑ 1827		

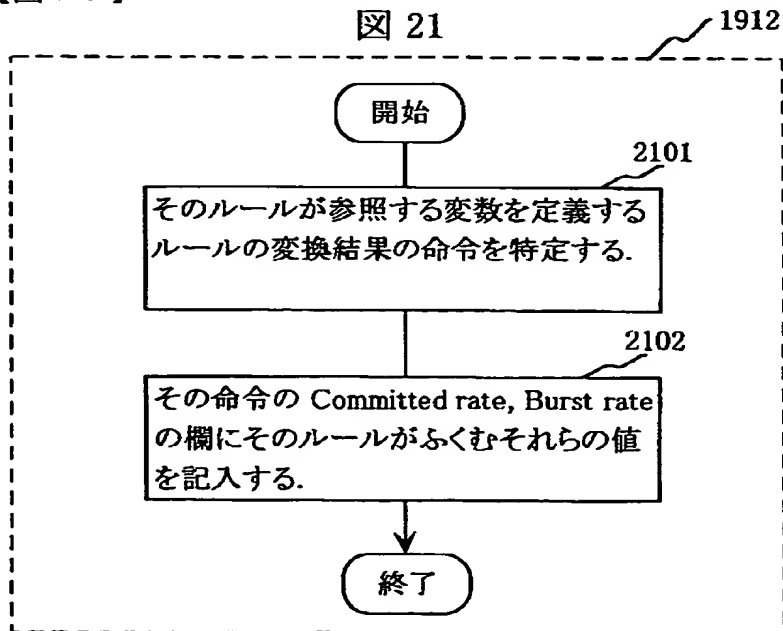
【図 19】



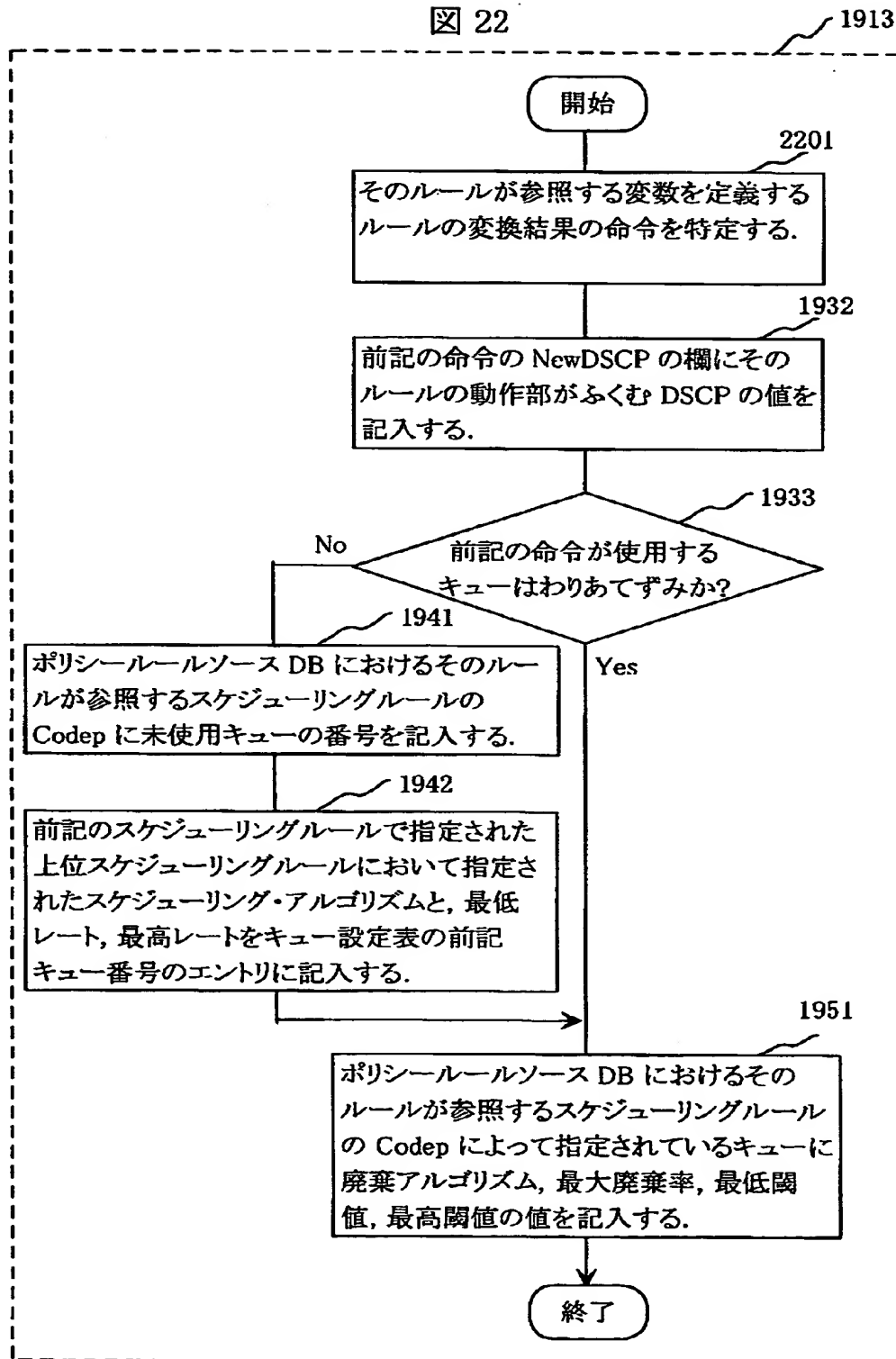
【図 2 0】



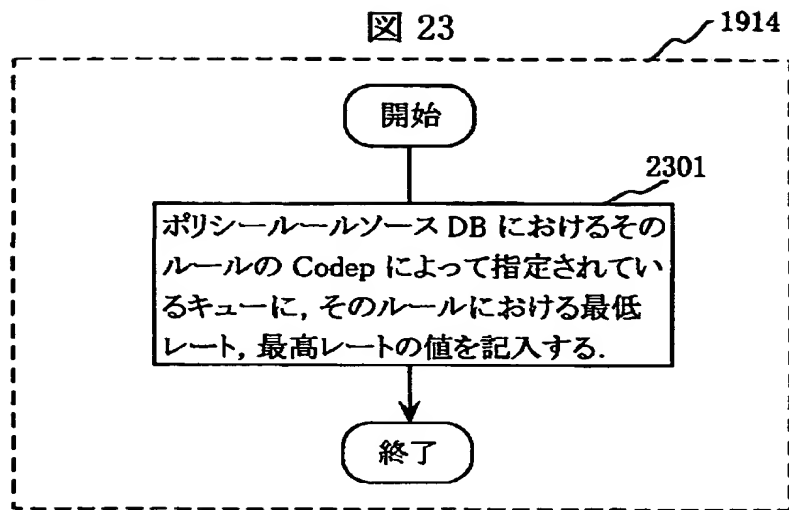
【図 2 1】



【図 22】



【図 2 3】



【図 2 4】

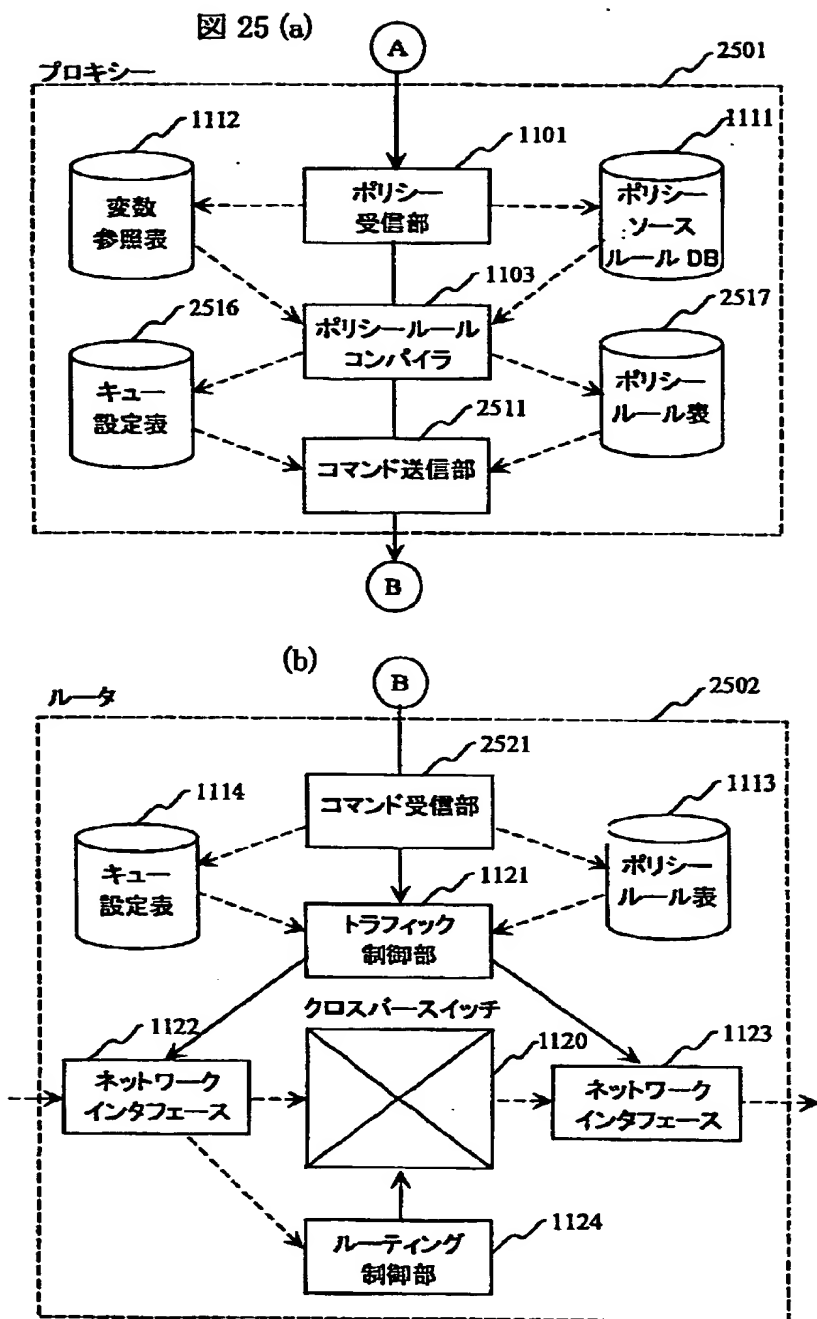
図 24

	2401	2402	2403	2404	
	↓	↓	↓	↓	
Op, RuleID, Interface	Deploy	#1	3		
ConditionLength	32				← 2412
Protocol	TCP				← 2413
SourceIPmin	192.168.4.1				← 2414
SourceIPmax	192.168.4.1				← 2415
SourcePort	null				← 2416
DestinationIPmin	null				← 2417
DestinationIPmax	null				← 2418
DestinationPort	null				← 2419
DSCP	null				← 2420
ActionLength	12				← 2421
NewLabel	1				← 2422
CommittedRate	1000				← 2423
BurstRate	null				← 2424

	2431	2432	2433	2434	
	↓	↓	↓	↓	
Op, RuleID, Interface	Redeploy	#1	3		

	2441	2442	2443	2444	
	↓	↓	↓	↓	
Op, RuleID, Interface	Undeploy	#1	3		

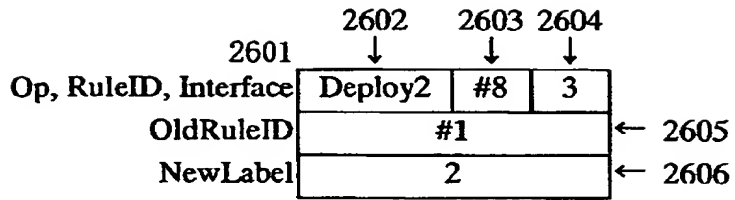
【図 25】



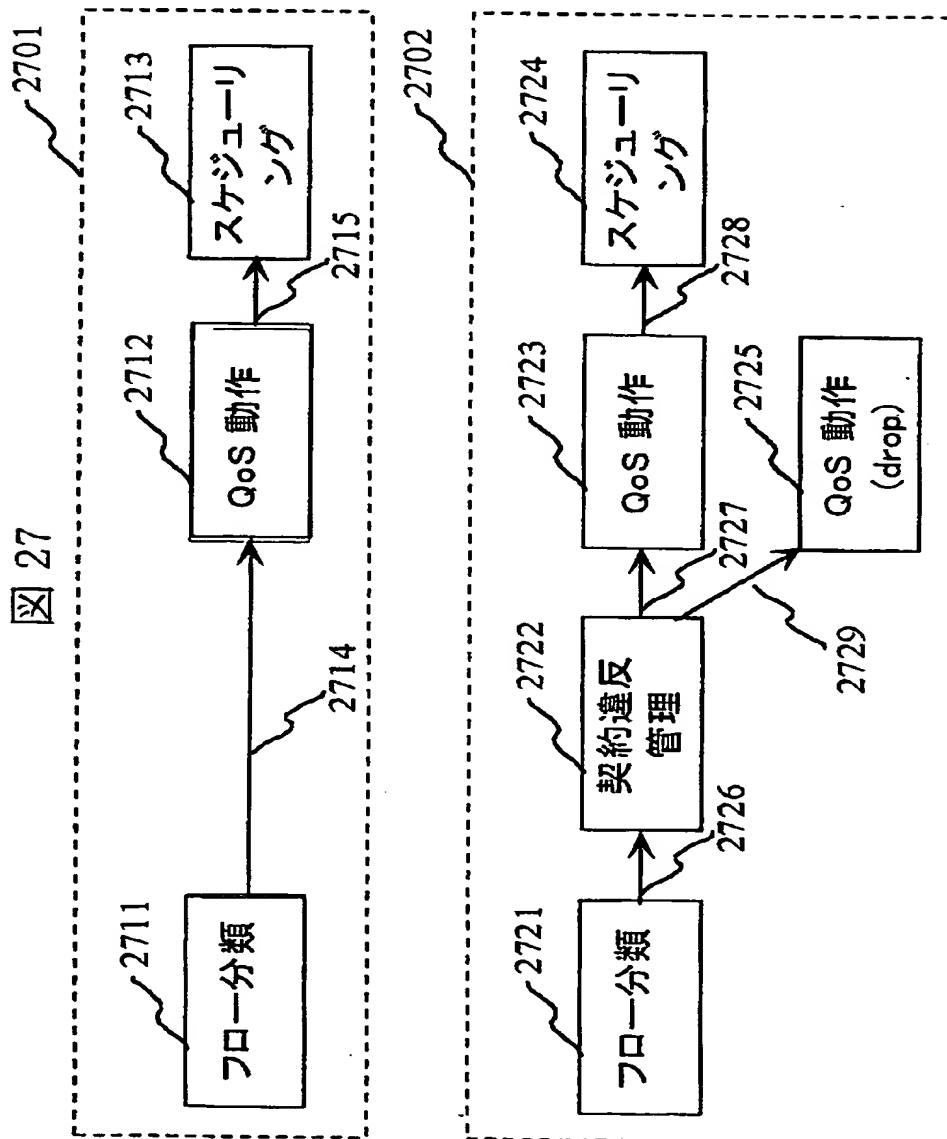


【図 2 6】

図 26



【図 2 7】



【書類名】 要約書

【要約】

【課題】 ポリシーベース・ネットワークにおいてネットワークの混雑を最小限におさえ、ダウンロード時間とポリシールールの変換にかかる時間を最小限にし、ポリシー制御が中断しないかまたは中断時間が最小限ですむようにし、またルータに過大な負荷がかかることがないようにすることを目的とする。

【解決手段】 ポリシールール間のデータの依存性を解析して最小限のポリシールールやデータのセットをもとめ、ポリシールールがルータに格納されているかどうかを判定してポリシーサーバがポリシールールの内容でなくその識別子だけを転送する。

【効果】 ネットワークの混雑が最小限におさえられ、ダウンロード時間およびポリシールールの変換にかかる時間が最小限になり、ポリシー制御が中断することがないかまたは中断時間が最小限ですむようになり、またルータに過大な負荷がかかることがない。

【選択図】 図 2

出 願 人 履 歴 情 報

識別番号 [000005108]

1. 変更年月日 1990年 8月31日

[変更理由] 新規登録

住 所 東京都千代田区神田駿河台4丁目6番地

氏 名 株式会社日立製作所